ESANN'2001 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 25-27 April 2001, D-Facto public., ISBN 2-930307-01-3, pp. 165-170

Motor control and movement optimization learned by combining auto-imitative and genetic algorithms.

Karl T. Kalveram & Ulrich Natke

Section Cybernetical Psychology und Psychobiology
Heinrich-Heine-University, Duesseldorf, Germany
email: kalveram@uni-duesseldorf.de

Abstract. In sensorimotor behaviour often a great movement execution variability is combined with a relatively low error in reaching the intended goal. This phenomenon can especially be observed if the limb chain under regard has redundant degrees of freedom. Such a redundancy, however, is a pre-requisit of movement optimization, because without variability changes in movement execution are impossible. It is, therefore, suggested, that, given a fitness criterion, a related optimal movement trajectory can be learned by an genetic algorithm. However, precise reaching must also be learned. This requires to establish at least an internal inverse model of the (forward) "tool transformation" governing the physical behaviour of the limb chain. Learning of an inverse model can be performed best applying the so called auto-imitation algorithm, a non-supervised learning mechanism equivalent to (modified) Hebbian learning. The paper shows theoretically, how these two learning algorithms can be combined in motor learning, and exemplifies by simulation of a three-jointed arm confined in a plane, how the problem of combining goal invariance under motor variability with movement optimization can be solved practically in a biologically plausible manner.

1. Introduction

Sensorimotor control has to guarantee that a perceptive goal is precisely hit. Because the human body and its limbs - regarded mechanically - usually have redundant degrees of freedom, the same goal can be reached by an infinite number of positional and/or speed trajectories. Observations reveal that our inbuild motor controller when repeatedly carrying out a goal directed movement is capable to achieve the goal, though every time another trajectory is selected. This phenomenon is known as 'goal invariance by motor variability'. It opens the possibility to select those trajectories that optimize the movement with respect to a distinct criterion, thus yielding for instance minimum metabolic energy consumption, minimum jerk, maximum convenience, or a straight line between beginning and ending of the movement, but preserving reaching precision. It is obvious that humans can behave in this manner, but it is not known how this is achieved. The purpose of the present paper is to demonstrate by an artificial system, what kind of control equipment is necessary to establish (a) goal invariance under movement variability, and (b) optimization of the movement execution with respect to different criterions.

ESANN'2001 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 25-27 April 2001, D-Facto public., ISBN 2-930307-01-3, pp. 165-170

2. Theory

A three-jointed arm confined in a plane (s. fig.1) is taken as an instance of a "tool" to be manipulated by the neural controller. The arm has three degrees of freedom, that is one more than necessary to reach every point in the working space.

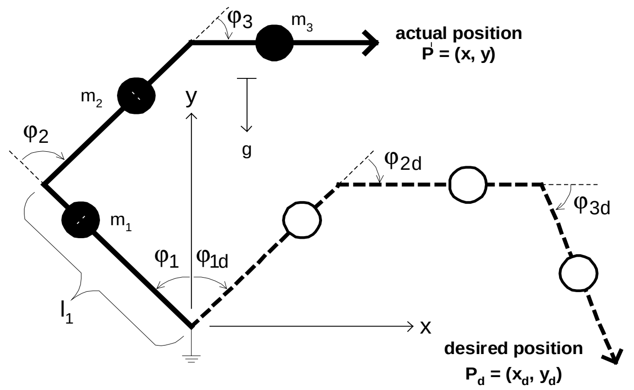


Figure 1: Three-jointed arm confined in a plane taken as the "tool" to be controlled. The position of the tip of the arm described in extero-ceptive - here Cartesian - co-ordinates is the effective part of this tool. The three joint angles represent the arm in proprioceptive co-ordinates.

The physical behaviour of the arm is described by the tool transformation that leads from the torques about the three joints to the position of the tip of the arm. The neural controller of the arm is assumed to provide an inverse model of the tool transformation: Given a desired position, the inverse model computes online those torques realizing that position. The tool transformation usually is partitioned into the (forward) dynamics handling with the inertia of the arm and leading from the joint torques to the joint angles, and into the (forward) kinematics handling with the geometrical properties of the limb chain and leading from the joint angles to the Cartesian position of the tip of the arm. Accordingly, the inverse model implemented in the controller can be split into the inverse dynamics and inverse kinematics. Assuming that the inverse dynamics exactly matches the forward dynamics, from Fig.1 it follows that the tool transformation simplifies to the kinematics expressed by

$$\begin{aligned} x &= l_1 \sin\varphi_1 + l_2 \sin(\varphi_1 + \varphi_2) + l_3 \sin(\varphi_1 + \varphi_2 + \varphi_3) \\ y &= l_1 \cos\varphi_1 + l_2 \cos(\varphi_1 + \varphi_2) + l_3 \cos(\varphi_1 + \varphi_2 + \varphi_3) \end{aligned} \qquad (1)$$

2.1 Acquisition of the inverse kinematics by auto-imitation

Inversion of the kinematics (1) can principally be performed by an algorithm I called "auto-imitation" (Kalveram 1992). As I pointed out previously (Kalveram 1981), this algorithm is related to the reafference principle (Holst & Mittelstaedt 1950) and also to Hebbian learning (Kalveram 2000), and can be characterized as a "direct method of inverse modelling" (Jordan 1988). In the present case, and in terms of cognitive psychology, the network trained by auto-imitation acquires the rule, which provides the angular values for a given position, by "inductive reasoning", that is to say, by showing some instances of that rule. Principle of operation is, that arbitrary angular values generated by a "blind teacher" are fed into the arm and into the network's learning input as well, whereas the positions attained by the tip of the arm are fed in the regular input of the network. After training, this will make the network's output specify the angles for any desired Cartesian position offered to the regular input.

Trying to invert (1) however conduces to an illposed problem, because there exists an infinitive number of combinations of angles realizing the same Cartesian position.

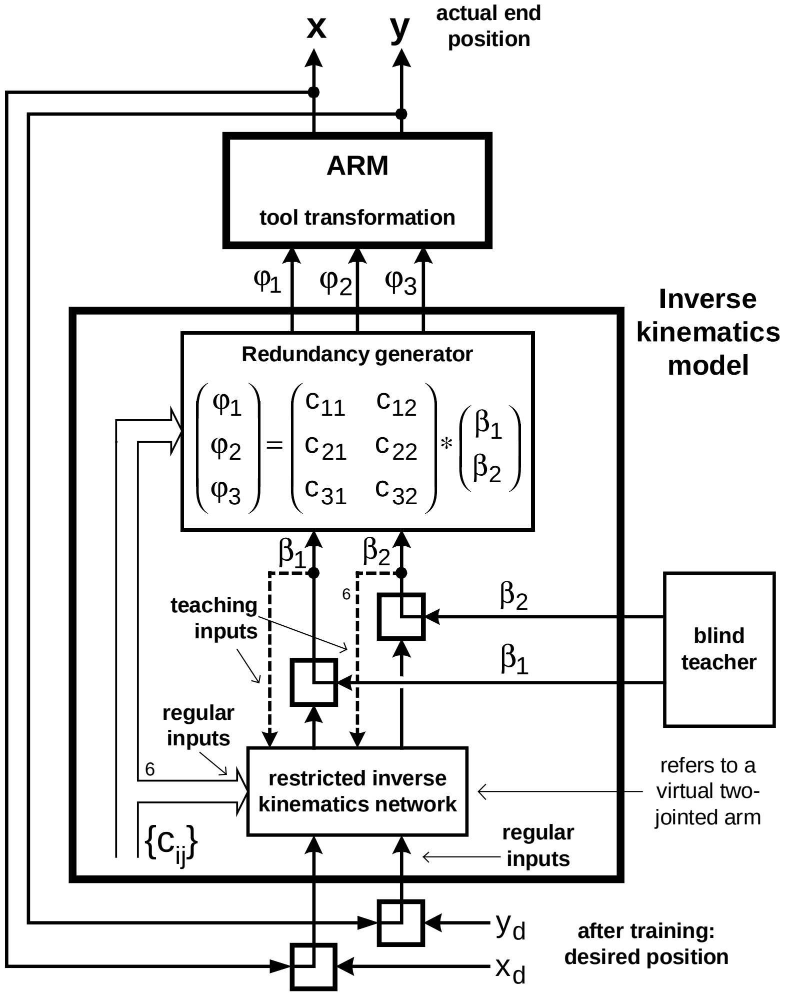


Figure 2: Neural controller learning the inverse kinematics of the arm of fig.1 in the framework of auto-imitation. See text for more information

Therefore, the controller must impose a rule on the joint angles making them dependent from each other in an explicit manner, and this rule has to be included into the training procedure. Fig.2 demonstrates how this can be managed in the framework of auto-imitation: In the learning phase, the redundancy generator containing the constraining rule gets a series of arbitrary pairs of angles from the "blind teacher". The blind teacher refers to a virtual non-redundant two-jointed arm and assumes, that this arm's inverse kinematics - in fig.2 denoted as restricted inverse kinematics - has to be learned. According to the constraining rule, the redundancy generator puts out three angles which are transformed into a Cartesian position by the tool transformation of the real arm. The Cartesian position is fed back via exteroception to two of the eight regular inputs of the network aquiring the restricted inverse kinematics. The constraining rule is formulated by a matrix operation, whereby the coefficients of matrix C – also called co-ordination matrix - must be provided in a descent manner. These co-ordination coefficients are fed into the redundancy generator and the six remaining regular inputs of the restricted kinematics network as well (double lined arrows in fig.2). The teaching inputs of this network are supplied also with the blind teacher's output. Therefore, the network gets all the information necessary to build a representation of the restricted inverse kinematics valid for all given sets of co-ordination coefficients. After learning is completed, the four switches (square boxes in fig.2) are brought into the alternative position. Now, the blind teacher is switched off, and a desired Cartesian position offered to the inverse kinematics unit will produce an angular output inducing the arm's tip to move to the desired position, thereby following the rule given by the coefficients of the actual co-ordination matrix.

In a simulation experiment an arm was used with the segment lengthes $l_1=l_2=l_3=1$ and the co-ordination coefficients $c_{11}=c_{22}=1$, $c_{12}=c_{21}=c_{31}=0$, $c_{32}=c$ (that is $\varphi_1=\beta_1$, $\varphi_2=\beta_2$, $\varphi_3=c\beta_2$). The network to be trained was a "power network" (Kalveram 1993). This is a three layer SIGMA-PI network with feedforward architecture, fixed synaptic weights in the hidden layer, and plastic weights in the output layer. Thereby, the hidden neurons multiply potentiated variables coming from the input layer, and the output neurons then compute a weighted sum of these products. If M=number of output neurons and K=number of input neurons, the power network represents M abbreviated K-dimensional power series, or even Taylor series, known to approximate any function to every required degree of precision. In the present investigation, such a power network was used, established with 8 input neurons, 26 hidden neurons, and 2 output neurons. The overall 52 weights of the output neurons can be interpreted as modified Hebbian synapses (Kalveram 2000). The 'updating rule' for the weigths was the simultaneous LSQ-rule (Kalveram, 1992, 1993), by which the outcome of each training trial was held in memory, until at the end of training all weights were determined at once by a least squares procedure.

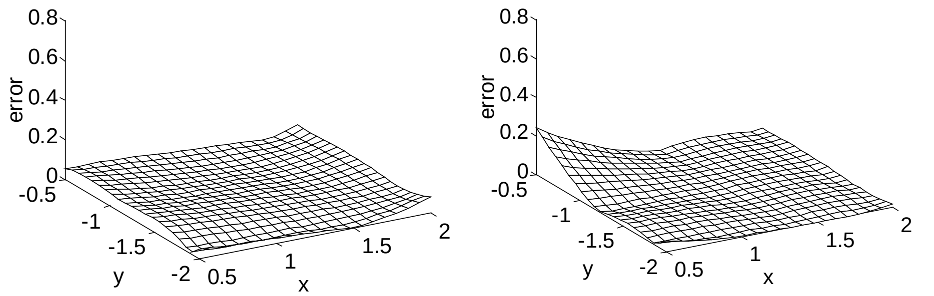
Auto-imitative training was accomplished as described above with 100 trials and c varying at random between 0 and 1. Results are shown in fig.3. They demonstrate the high effectivity of the auto-imitation procedure when acquiring the capability to hit a goal precisely though following different trajectories leading to that goal.

2.2 Movement optimizing by a genetic algorithm

The solution of the redundancy problem worked out above implies also the solution of the optimizing problem, because the joint angles put out by the redundancy generator can principally be chosen such that a predefined secondary optimality criterion, given for instance as $F(\varphi_1,\varphi_2,\varphi_3)\Rightarrow\min$, is approached as closely as possible. However, at the beginning it is unknown what set of co-ordination coefficients must be selected to meet the criterion.

ESANN'2001 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 25-27 April 2001, D-Facto public., ISBN 2-930307-01-3, pp. 165-170

Performance test with c=1 | Performance test with c=0,5



e = 0.0310, s = 0.0166 | e = 0.0225 , s = 0.0337

Figure 3: Relative errors for **c=0.5 and c=1** in dependency of the Cartesian goal position when testing the inverse kinematics learned by auto-imitation. The targets x,y to be hit by the tip of the arm are given by the 441 points of a test grid (grid constant =0.075) covering the working area of the arm. Error is defined as absolute distance between the goal and the actually reached position of the tip of the arm, divided by half the diagonal of the working area. e denotes the average error, and s the standard deviation of the errors yielded for the points of the test grid. All computations were made with MATLAB.

Fig.4 visualizes how, based on a backpropagation network, suitable co-ordination coefficients can be found using a genetic algorithm. Having the inverse kinematics model established, the procedure starts with arbitrary co-ordination coefficients put out by the network. Prior to movement start, noise is added to these coefficients changing them a little bit at random. After performing the movement the value of the criterion of the current trial is compared to that of the previous trial. Each time the current value is better, the synaptic weights of the network are adjusted with the changed coefficients as target values.

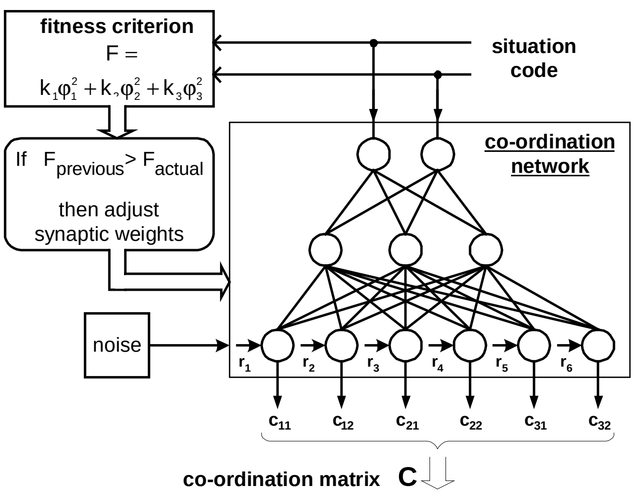


Figure 4: Movement optimizing by a genetic algorithm. The situation code selects the "fitness-criterion" to be applied. The function F in the box at top serves as an example. k_1,k_2,k_3 are fixed parameters in F. The situation code also induces the network to put out the related co-ordination coefficients , which are superimposed by noise for the purpose of further optimization.

In a simulation, the criteria were chosen to $F = k_1\varphi_1^2 + k_2\varphi_2^2 + k_3\varphi_3^2 \Rightarrow$min with k_2=1 and (a) k_1=4, k_3=1 or (b) k_1=1, k_3=4. This may map the situations where a person is martyred by rheumatism in the shoulder (a) or in the wrist (b) and therefore is motivated to minimize rotation about that joint. Repeating the procedure outlined in fig.4 while switching between situations a and b indeed gets values for φ_1 respectively φ_3 considerably lesser on average than for the other angles.

3. Discussion

The controller model outlined solves the problem of handling an arm with redundant degrees of freedom by introducing (1) a non-redundant inverse kinematic model for the arm, (2) a redundancy generating network containing a linear constraining rule bending the joint angles, and (3) a co-ordination network providing the coefficients actually co-ordinating the movements of the arm segments. Applying auto-imitation establishes the capability to reach a goal while movement variability is preserved. This in turn is a pre-requisit that movement optimization can take place using an genetic algorithm. Principally, the linear constraining rule can be replaced by a non-linear one. That will enhance the number of criterions available for movement optimization. Whether the suggested control mechanism applies also to non-linear constraining rules is subject to further research.

Acknowledgements: (1) The study was supported by grants Ka417/18-3 and Ka417/13-3 from Deutsche Forschungsgemeinschaft (DFG).
(2) The authors thank Tony Illenberger for implementing the genetic algorithm.

References

Holst,E. von & H. Mittelstaedt, 1950. Das Reafferenzprinzip. Naturwissenschaften 37, 464-476

Jordan, M. I., 1988. Supervised learning and systems with excess degrees of freedom. COINS Technical Report 88-27, 1-41

Kalveram, K.Th., 1981. Erwerb sensumotorischer Koordinationen unter störenden Umwelteinflüssen: Ein Beitrag zum Problem des Erlernens von Werkzeuggebrauch. (Engl. titel: Acquisition of sensorimotor co-ordinations under environmental disturbations. A contribution to the problem of learning to use a tool) In: L. Tent (Ed.): Erkennen, Wollen, Handeln. Festschrift für Heinrich Düker (S. 336-348). Göttingen: Hogrefe

Kalveram, K.Th., 1992. A neural network model rapidly learning gains and gating of reflexes necessary to adapt to an arm's dynamics. Biological Cybernetics 68, 183-191

Kalveram K.T., 1993. Power series and neural-net computing. Neurocomputing 5, 165-174

Kalveram, K.T., 2000. Sensorimotor Sequential Learning by a Neural Network Based on Redefined Hebbian Learning. In H. Malmgren, M. Borga & L. Niklasson (Eds) Artificial Neural Networks in Medicine and Biology. London: Springer