ESANN'2008 proceedings, European Symposium on
Artificial Neural Networks - Advances in Computational Intelligence and Learning
Bruges (Belgium), 23-25 April 2008, d-side publi., ISBN 2-930307-08-0.

Combining neural networks and optimization techniques for visuokinesthetic prediction and motor planning

Wolfram Schenck, Dennis Sinder, and Ralf Möller

Computer Engineering Group - Faculty of Technology
Bielefeld University - POB 100131, D-33501 Bielefeld - Germany

Abstract. We present a method for motor planning based on visuokinesthetic prediction by a forward model (FM) and the optimization method "differential evolution" (DE) for a block-pushing task of a robot arm. The FM is implemented by a set of multi-layer perceptrons and used for the iterative prediction of future sensory states in an internal simulation process. DE is applied to determine via this internal simulation the movement sequences by which a target block can be successfully pushed from an arbitrary start to an arbitrary goal position. The presented method shows a good performance on the pushing task.

1 Introduction

In the field of embodied cognitive science, it is a hypothesized that perception and cognition rely on internal simulation processes which involve neural structures for motor control [1, 2, 3, 4]. Internal simulation requires forward models (FMs) which predict the sensory consequences of motor actions. These predictions can be used for an iterative simulation of sequences of motor commands, a process closely related to motor planning. Internal simulation of this kind faces two main problems: First, the FM has to be precise enough to allow for an iterative prediction without ending up in a completely erroneous final state. Second, one has to find a way to avoid the combinatorial explosion which occurs if several motor commands are tested in parallel at each iterative prediction step.

In the present study, we develop a procedure to deal with both problems and demonstrate its performance for the task of visually guided block-pushing with a robot arm. On the one hand, we offer a technical solution for motor planning based on multi-layer perceptrons [5] and the optimization method "differential evolution" [6], on the other hand, our model can be interpreted as an instance of simulation theories of visual perception [3]. However, the second aspect is beyond the scope of this paper, and we would like to refer the reader to [7] for further information on this topic. Our optimization approach to motor planning is related to the work by Hoffmann [2] (who used "simulated annealing" for the generation of movement sequences for a mobile robot) and by Tani [8] (who used "chaotic steepest descent" for a similar purpose).

2 Setup and task

The used robot arm setup and the world coordinate system are shown in Fig. 1.

ESANN'2008 proceedings, European Symposium on
Artificial Neural Networks - Advances in Computational Intelligence and Learning
Bruges (Belgium), 23-25 April 2008, d-side publi., ISBN 2-930307-08-0.

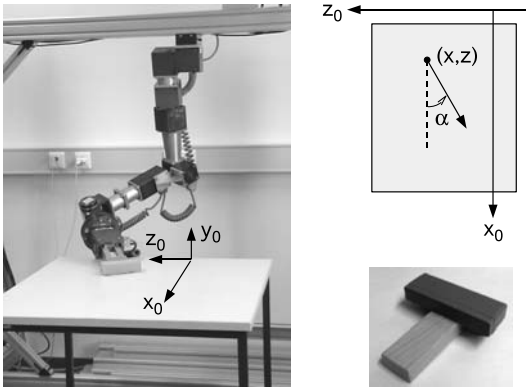

Fig. 1: Left: The robot arm in a pushing posture with the block in front of the gripper. Upper right: Base coordinate system on the table surface (see also left picture). The working area for pushing movements is shown in gray color. A robot arm posture is defined by the gripper position (x, z) and the pushing orientation α. Lower right: Tool held by the gripper during pushing [7].

For the block-pushing task presented in this study, movements of the arm are restricted to a 2D plane at the white table surface. With the help of a special tool held in its gripper, the robot arm pushes a small block around this surface. The posture of the robot arm is defined by the workspace coordinates x and z of the gripper tip and by an angle α indicating the pushing orientation. The remaining degrees of freedom are fixed, resulting in robot arm postures as shown in Fig. 1 (left). Collision-free operation is only possible for a restricted area of the table surface defined by $x \in [330\,\text{mm}; 730\,\text{mm}]$ and $z \in [-69.5\,\text{mm}; 250.5\,\text{mm}]$ ($\alpha \in [-40°; +40°]$). Visual data is collected with a camera that records the entire white table surface.

The task of the robot arm is to push the block from a start position to a goal position within the working area (with varying orientations). The general pushing direction is directed away from the base joint. The block is first placed at its goal position by the operator and afterwards at its start position. At both positions, a camera image is recorded. From these images, a sequence of motor commands is determined by which the robot arm manages to push the block from the start to the goal. To generate this sequence by an internal simulation process, a visuokinesthetic FM is required. The FM predicts visual data (position and orientation of the block in the camera image) and kinesthetic data (position and orientation of the gripper as indicators of the arm posture) resulting from a given movement. Visual prediction is a difficult task because of the high dimensionality of visual data. For this reason, we drastically reduced its dimensionality. This is possible since we only have to encode the position and orientation of the block.

First, the camera image is converted into a monochrome image in which

all pixels of the block get maximum intensity and all other pixels zero intensity. From this image, a lowpass-filtered and subsampled version with only 3×3 pixels is created. The resulting 9 pixel intensity values encode the position of the block. The orientation of the block is encoded by the four values of a compass filter histogram. Four compass filters enhance the edges of the block segment in the full-size monochrome image in four different directions (0°, 45°, 90°, and 135°). After thresholding, the remaining pixels in each image are counted to give a value for the distribution of edges in a given direction [9].

3 Network structure and training

The visuokinesthetic FM for the internal simulation process has the following inputs: First, the current gripper position and orientation as kinesthetic input $\mathbf{s}_{\text{KIN}}^{(t)} = (x_t, z_t, \alpha_t)$ (t denotes the time step); second, a 13-dimensional vector $\mathbf{s}_{\text{VIS}}^{(t)} = (\mathbf{s}_{\text{POS}}^{(t)}, \mathbf{s}_{\text{OR}}^{(t)})$ comprising the 3×3 pixel intensities encoding the block position $\mathbf{s}_{\text{POS}}^{(t)}$ and the four values of the compass filter histogram $\mathbf{s}_{\text{OR}}^{(t)}$; and third, a motor command $\mathbf{m}_t = (\Delta x_t, \Delta z_t, \Delta \alpha_t)$. The output of the FM consists of the visuokinesthetic state of the next time step, encoded by $\widehat{\mathbf{s}}_{\text{KIN}}^{(t+1)}$ and $\widehat{\mathbf{s}}_{\text{VIS}}^{(t+1)}$.

Learning this relationship is a function-approximation task; for this reason, the FM is implemented by a set of multi-layer perceptrons (MLPs) [5]. 37500 learning examples for the MLPs were generated by systematically moving the gripper of the robot arm along different trajectories through the working area while it was pushing the block. The movements were either translations in the current gripper direction α of a size of 10, 20, or 30 mm or rotations by a small angle $\Delta\alpha = 5°$. At each movement step, a full learning example was collected.

The best prediction performance was obtained by three separate MLPs for each output $\widehat{\mathbf{s}}_{\text{KIN}}^{(t+1)}$, $\widehat{\mathbf{s}}_{\text{POS}}^{(t+1)}$, and $\widehat{\mathbf{s}}_{\text{OR}}^{(t+1)}$, with the already described input encoding and pattern set size, and with plain online backpropagation as learning algorithm [5]. Each MLP of the visuokinesthetic FM has a single hidden layer with 10 units with hyperbolic tangent as activation function. After 300 epochs of network training with normalized data, we obtained the following prediction accuracy on a test set: The average absolute percentage difference between the correct output and the network-generated output amounts to less than 3% for the position and orientation output units and to nearly 0% for the kinesthetic output units (the percentages are computed in relation to the desired output).

4 Motor planning as optimization problem

The internal simulation process for motor planning requires an iterative application of the visuokinesthetic FM. For $t = 1$ with known sensory input $\mathbf{s}_1 = (\mathbf{s}_{\text{KIN}}^{(1)}, \mathbf{s}_{\text{VIS}}^{(1)})$, an adequate motor command $\mathbf{m}_1$ has to be generated (without executing it). The FM predicts the sensory state $\widehat{\mathbf{s}}_2 = (\widehat{\mathbf{s}}_{\text{KIN}}^{(2)}, \widehat{\mathbf{s}}_{\text{VIS}}^{(2)})$ of the next time step $t = 2$, a second motor command $\mathbf{m}_2$ is generated (without execution), the FM predicts the sensory state for $t = 3$ on the basis of the input $(\widehat{\mathbf{s}}_2, \mathbf{m}_2)$,

ESANN'2008 proceedings, European Symposium on
Artificial Neural Networks - Advances in Computational Intelligence and Learning
Bruges (Belgium), 23-25 April 2008, d-side publi., ISBN 2-930307-08-0.

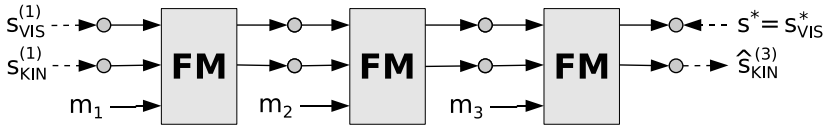

Fig. 2: The iterative application of the visuokinesthetic FM, depicted exemplary as chain of three FMs. The initial sensory state is used as input to the chain, the final output $\widehat{\mathbf{s}}^{(3)}_{\text{VIS}}$ of the last FM is constrained to be as close to the desired sensory goal state $\mathbf{s}^*$ as possible (indicated by the left-pointing arrow from $\mathbf{s}^*$).

and repeatedly so, until the number of prediction steps is equal to a predefined maximum N. Such an iterative application of an FM is shown in Fig. 2.

In the block-pushing task, the initial sensory state $\mathbf{s}_1 = (\mathbf{s}^{(1)}_{\text{KIN}}, \mathbf{s}^{(1)}_{\text{VIS}})$ is determined from the initial posture of the robot arm (ready to push the block from the start position) and the camera image showing the block at this position. The sensory goal state $\mathbf{s}^* = \mathbf{s}^*_{\text{VIS}}$ is determined from the camera image that shows the block at its goal position. It is important to note that $\mathbf{s}_{\text{KIN}}$ is *not* part of the sensory goal state. The system has no direct way to determine the kinesthetic state at the goal position. A movement sequence $\{\mathbf{m}_t\}$ is successful if the difference between $\widehat{\mathbf{s}}^{(N)}_{\text{VIS}}$ and $\mathbf{s}^*$ is very small. If $\{\mathbf{m}_t\}$ is actually executed afterwards, the final real sensory state $\mathbf{s}^{(N)}_{\text{VIS}}$ may differ considerably from $\mathbf{s}^*$, depending on the precision of the prediction by the visuokinesthetic FM. Thus, a precise FM is an important precondition for a realistic internal simulation process.

The optimization problem for the generation of a movement sequence is stated as follows. The initial sensory state is given by $\mathbf{s}_1 = (\mathbf{s}^{(1)}_{\text{KIN}}, \mathbf{s}^{(1)}_{\text{VIS}})$, the sensory goal state by $\mathbf{s}^* = \mathbf{s}^*_{\text{VIS}}$. The number of iteration steps is set to a fixed number N. The optimization goal is the minimization of the difference between $\widehat{\mathbf{s}}^{(N)}_{\text{VIS}}$ and $\mathbf{s}^*$. The free parameters in the optimization process are the motor parameters in the sequence $\{\mathbf{m}_t\}$ $(t = 1..N)$. They are constrained such that translatory movements are only simulated in direction of the respective current gripper orientation (like in the training data).

Differential evolution (DE) [6], an evolutionary optimization algorithm, is used as optimization method with a population size of $N_{\text{DE}} = 50$ and a maximum number of $G_{\max} = 15$ generations (for further details see [7]). The energy E which indicates the fitness of a movement sequence (the smaller E the better) is computed by a criterion which defines a tradeoff between position and orientation accuracy. Moreover, penalty terms are added to E if any motor parameter m_t is outside the range that the MLPs of the visuokinesthetic FM have encountered during training, or if any estimated kinesthetic state $\widehat{\mathbf{s}}^{(t)}_{\text{KIN}}$ during the simulation of the movement sequence is outside the working area.

Since the distance between the start and the goal is not known beforehand, the optimization process has to be carried out with different numbers of iteration steps N. We varied N between 7 and 15. Considering the population size

ESANN'2008 proceedings, European Symposium on
Artificial Neural Networks - Advances in Computational Intelligence and Learning
Bruges (Belgium), 23-25 April 2008, d-side publi., ISBN 2-930307-08-0.

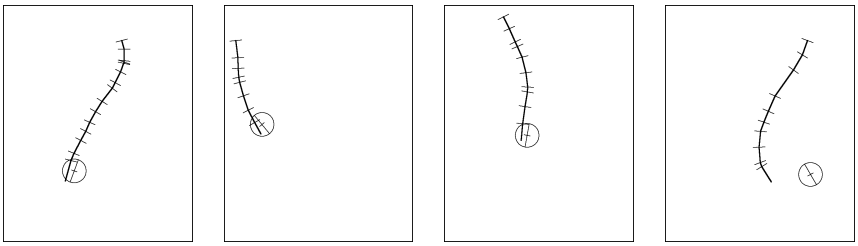

Fig. 3: Simulated trajectories for 4 different start and goal positions (for details see text). The figures underneath each trajectory indicate the final position error (left; in mm) and the final orientation error (right; in degrees).

$N_{\mathrm{DE}} = 50$ and the maximum number of generations $G_{\max} = 15$, the computation of the best movement sequence required the internal simulation of 6750 different movement sequences. The sequence which resulted from the optimization trial with the lowest final energy E was picked as overall best movement sequence. However, for a fair comparison between optimization trials with a different iteration depth N, we multiplied E before the comparison with an "increase factor" of 1.2^N. This is motivated by the fact that the precision of the final prediction gets worse the more internal simulation steps have to be carried out.

5 Results

The results that are reported here were generated in an experiment in which 100 movement tasks with different random start and goal positions were solved. Certain constraints were applied to these randomly generated movement tasks to ensure that they are geometrically possible, and that the overall orientation difference is not too large. For each movement task, the optimization process generated a movement sequence by the algorithm described in the preceding section. By executing this sequence, the block would have been ideally pushed to the goal position which is encoded by $\mathbf{s}^* = \mathbf{s}^*_{\mathrm{VIS}}$. The corresponding desired final arm posture is denoted as $\mathbf{s}^*_{\mathrm{KIN}} = (x^*, z^*, \alpha^*)$, the actual final arm posture after the movement as $\mathbf{s}^{(N)}_{\mathrm{KIN}} = (x_N, z_N, \alpha_N)$.

The mean position error, defined as the Euclidean distance between (x_N, z_N) and (x^*, z^*), amounted to 27.9 mm for the 100 movement tasks ($\sigma = 18.2$ mm), the mean orientation error, defined as absolute value of the difference between α_N and α^*, to 6.0 degrees ($\sigma = 6.3$ degrees). For the mean movement distance, defined as the distance between the start and the goal position of a movement task in the (x, z) space, we obtained 175 mm. The percentage ratio between the mean position error and the mean movement distance was 16.0%. On average, a movement sequence had a length of 8.6 steps. The movement distance was correlated to the length of the corresponding movement sequence ($r = 0.33$) and to the resulting position error ($r = 0.73$).

Figure 3 shows the movement sequences that were generated in 4 of the 100 movement tasks. Each panel depicts the complete working area, the x-axis pointing in the vertical direction, the z-axis in the horizontal direction. The goal position (x^*, z^*) is indicated by a circle with a diameter of 20 mm in each panel. The longer bar of the cross that marks the center of the circle points into the goal orientation α^*. Successive movement steps within a sequence are separated by bars that are orthogonal to the movement direction. The first three examples show rather precise solutions over various movement distances, while the example on the right illustrates a failed solution.

6 Conclusions

We presented a method for movement planning for a block-pushing task based on visuokinesthetic prediction and on optimization by DE. The performance of our method is rather good, even movement sequences with a length of 15 steps are generated with tolerable final position and orientation errors (leftmost trial in Fig. 3). However, success is not always guaranteed. The most likely reason for failed trials is a sub-average prediction accuracy of the visuokinesthetic FM in some regions of its input space. While the kinesthetic prediction by the FM is nearly flawless, the visual position and orientation data is predicted with an average accuracy of 3%. This sounds tolerable, but even small prediction errors accumulate easily, rendering the final output useless. This is reflected by the strong correlation between movement distance and final position error. Thus, further research has to focus on the training of even more precise FMs.

References

[1] H.-M. Gross, A. Heinze, T. Seiler, and V. Stephan. Generative character of perception: A neural architecture for sensorimotor anticipation. *Neural Networks*, 12:1101–1129, 1999.

[2] Heiko Hoffmann. *Unsupervised Learning of Visuomotor Associations*. MPI Series in Biological Cybernetics. Logos Verlag, Berlin, 2004.

[3] Ralf Möller. Perception through anticipation — a behavior-based approach to visual perception. In A. Riegler, M. Peschl, and A. von Stein, editors, *Understanding Representation in the Cognitive Sciences*, pages 169–176. Plenum Academic, New York, 1999.

[4] Tom Ziemke, Dan-Anders Jirenhed, and Germund Hesslow. Internal simulation of perception: A minimal neuro-robotic model. *Neurocomputing*, 68:85–104, 2005.

[5] D. E. Rumelhart, G. Hinton, and R. Williams. Learning internal representations by error propagation. In D. E. Rumelhart and J. L. McClelland, editors, *Parallel distributed processing: Explorations in the microstructure of cognition. Vol. 1: Foundations*, pages 318–362. MIT Press, Cambridge, MA, 1986.

[6] Rainer Storn and Kenneth Price. Differential evolution — a simple and efficient heuristic for global optimization over continuous spaces. *J. of Global Opt.*, 11:341–359, 1997.

[7] Dennis Sinder. Roboterarm-Ansteuerung mit Hilfe von visuellen Vorwärtsmodellen, 2006. Diploma Thesis. Computer Engineering Group, Faculty of Technology, Bielefeld Univ.

[8] Jun Tani. Model-based learning for mobile robot navigation from the dynamical systems perspective. *IEEE Transact. on Systems, Man, and Cyb. — Part B*, 26:421–436, 1996.

[9] Heiko Hoffmann, Wolfram Schenck, and Ralf Möller. Learning visuomotor transformations for gaze-control and grasping. *Biological Cybernetics*, 93:119–130, 2005.