Neural Network Approaches for Sensory-Motor-Coordination

Sensormotorische Koordination mit Neuronalen Netzen

Helge Ritter, Technische Fakultät; Holk Cruse, Fakultät für Biologie, Universität Bielefeld

Prof. Dr. H. Ritter leitet seit 1990 die Abteilung „Mustererkennung und Verarbeitung" an der Technischen Fakultät der Universität Bielefeld.

Prof. Dr. H. Cruse leitet seit 1981 die Abteilung „Biologische Kybernetik" an der Fakultät für Biologie der Universität Bielefeld.

Sensor-based coordination of movements is a central task for artificial robots and biological organisms as well. While traditional algorithms have largely relied on rather detailed models of the kinematics and dynamics of this process, neural networks offer the possibility to replace a significant amount of modeling by adaptation and learning. Moreover, principles of movement coordination observed in biological organisms can be used to construct networks exploiting these principles for the control of artificial devices. In this contribution, we will report on some work that adresses both issues and that is part of a larger, interdisciplinary research effort aiming at the construction of a neural-network controlled robot system.

Die sensorbasierte Steuerung von Bewegungen bildet eine zentrale Aufgabe für Roboter und biologische Organismen gleichermaßen. Während traditionelle Algorithmen in erster Linie auf vergleichsweise detaillierten Modellen der Kinematik und der Dynamik des Bewegungsvorgangs beruhen, bieten Neuronale Netze die Möglichkeit, einen erheblichen Teil dieser Modellierung durch Adaptation und Lernen zu ersetzen. Darüberhinaus können in der Natur beobachtete Bewegungskoordinationsmechanismen als Grundlage für das Design von Netzwerken zur Steuerung künstlicher Systeme dienen. In diesem Beitrag soll über Arbeiten berichtet werden, die beide Gesichtspunkte zum Gegenstand haben, und die Bestandteil eines größeren, interdisziplinären Forschungsprojekts sind, das die Realisierung eines durch Neuronale Netzwerke gesteuerten Robotersystems verfolgt.

1. Introduction

Carrying out skillful movements is a difficult task. This statement is not supported by our everyday experience; owing to the superb performance of the motor systems in our brain we can appreciate the involved complexities only when trying to program artificial robots ourselves (see, e.g. Brady 1989).

Programming robots can be viewed as a problem of knowledge acquisition. Much of the required knowledge concerns laws that can coordinate the movements of multiple joints, taking into account a complex context provided by a variety of sensory signals of tactile or visual origin (Hildreth and Hollerbach 1985). This points out an important difference to the knowledge acquisition problem faced by traditional knowledge based systems: in robotics, a major part of the relevant knowledge concerns inherently continuous quantities, such as sensor signals and joint torques. A second difference may at first glance seem extremely favorable: since the analysis of movements and of mechanical interactions can be based on the firm foundations of classical physics, one might hope that the problem of knowledge acquisition can to a large extent be bypassed by a mathematical analysis of the intended operations, something that usually cannot even be attempted in many other domains of interest. However, while extremely powerful in principle, such analytic approach is severely limited in practice. Most situations of interest are simply too complicated to be amenable to analysis at affordable costs. Imagine typical everyday tasks such as opening a button, walking over a pile of stones, picking a candy from a box, or manipulating spaghetti with a fork. Developing an analytical description for each of these tasks constitutes a daunting problem, yet many of the tasks we would like to delegate to robots are precisely of this type. One should note that the difficulty of these tasks is not rooted in issues of planning or sophisticated reasoning. Instead, the main source of their difficulty is the need to coordinate a large number of different mechanical degrees of freedom according to very complex sensory feedback signals from vision and from tactile and force sensors (Brooks 1990).

In view of the practical limitations of the analytical approach we must develop methods to complement it. Here, we advocate two additional sources of knowledge: the first is to develop good methods for robot learning. Since we all are experts at carrying out movements, good robot learning algorithms will greatly facilitate robot programming. However, we should not attempt to start with a "tabula rasa". To maximally exploit the potential of learning, we should try to identify generic types of movement patterns and control strategies that then need only be refined by learning. Biology offers a rich reservoir of such information. Natural motor systems employ a variety of different movement coordination and reflex patterns that have been optimized over millions of years (see, e.g. Cruse 1990). Analysing these patterns and their underlying laws of coordination can provide an extremely valuable basis for technological approaches.

For research on learning algorithms and on biological motor control strategies a most natural framework seems to be provided by neural networks (see, e.g., Rumelhart and McClelland 1985, Grossberg and Kuperstein 1986, Hertz et al. 1991, Ritter et al. 1991). Besides their obvious relations to both learning and biology, these systems have a couple of additional attractive features: they are well suited to represent both continuous and discrete, symbolic quantities; they offer simple mechanisms to achieve noise resistance; they can be taylored to work with imprecise inputs and they are inherently parallel. These are compelling reasons to investigate neural network-based strategies for movement coordination and robot programming (Brüwer and Cruse 1990, Martinetz et al. 1989, Kawato 1987, Ritter et al. 1991). As concrete examples, this contribution reports on work that is focused on two important domains: the generation and control of hand and of walking movements. Both domains adress the important issue of multi-limb coordination under sensory feedback, but at different levels and with different aspects in the foreground. In the case of hand movements, the focus is primarily on the use fo visual feedback signals to control the grasping process and it is chiefly the complex kinematics of the interaction between hand and object that must be controlled. In the case of the walking movements, the focus is on how to control multiple limbs in a synchronized fashion to achieve various gaits and to take the properties of the ground into account. Here, kinematics alone is no longer sufficient, and dynamic aspects also need to be considered.

Both projects are part of a larger, interdisciplinary research effort and are "bracketed" by cooperating research projects studying on the biological side the neural basis of vision in anurans, such as toads and frogs, and by a project dedicated to the development of an articulated robot hand on the technological side. The aim of this larger, joint effort is to evaluate the potential of neural networks for the realization of biologically inspired robot control strategies and to demonstrate their feasability by an actual implementation in an important practical domain.

2. Motor Learning and the Problem of Dimensionality

One major obstacle in the control of complex movements, such as grasp movements, seems to be the high dimensionality of the configuration space involved. Leaving aside any complexities arising from different object shapes, a human hand has of the order of 16 degrees of freedom. If we attribute to each degree of freedom the moderate number of five different independent positions, we arrive at 5^{16} or more than 10^{11} different hand configurations. Of these, we can at best only explore a tiny fraction during our human lifetime of considerably less than 10^{10} seconds. In fact, a considerable range of different hand postures can be generated even with "convex combinations" of only three different hand postures. Figs. 1a–f show a few examples for a simulated robot hand of 12 degrees of feedom: the first three images (Fig. 1a–c) show a stretched

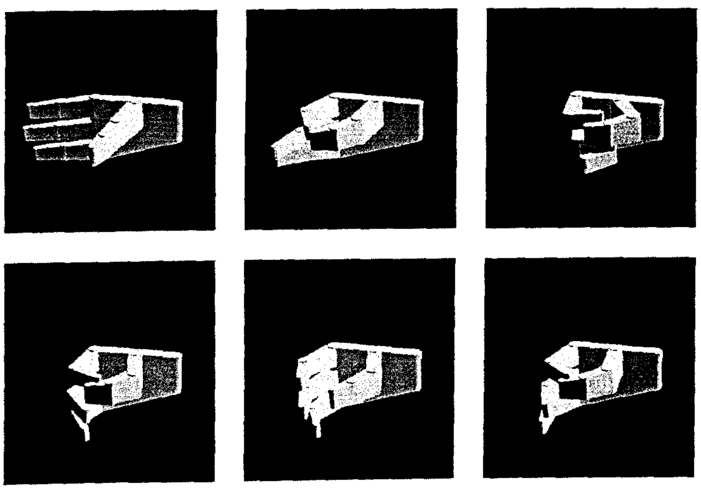

Fig. 1a–f Top row (a–c): three "basis postures" used to parametrize configurations of a 12-jointed robot hand. Bottom row (d–f): some examples of further configurations obtained by linear combination of the three postures shown in Figs. (a–c).

hand, a "precision grip" and a fist, which may be used as a rather versatile set of "basis postures" from which many further configurations, such as those shown in Figs. 1d–f, can be derived by linear combination. Any such combination involves only three coefficients (in our case, convex combinations were used, i.e. the coefficients were further constrained to sum to unity, so that only two degrees of freedom were involved), thus projecting the vast configuration space of 12 degrees of freedom of the full hand to a much more manageable space of three or even two dimensions. Of course, certain configurations cannot be realized in this way. However, for many tasks a sufficiently similar posture may be found among the linear combinations, and for the remaining actually occurring cases a special configuration may be easily stored, which then may serve as a "center" for a whole new submanifold of postures obtained by admixture of the previous three basis hand postures.

Similarly, actually occurring sensor signals range only over a tiny subset of the full set of their combinatorially possible combinations. Often, this subset can be well described by a fairly restriced number of "prototypes", so that the overlaps of the actual sensor readings with these prototypes provide sufficiently accurate information to e.g. make feedback adjustments to an ongoing movement. Below, we shall demonstrate the feassibility of this approach for the task of reconstructing the 3d-posture of the hand shown in Fig. 1 from 2d-pixel images.

In this way, we can both encode sensory input and motor output quantities for a robot, using only low-dimensional, approximate descriptions. These approximate descriptions can then be manipulated much more easily than the original, exact descriptions that involve all degrees of freedom that are potentially available. The price to pay is a limited accuracy; however most of our movement skills are not a result of a particularly high degree of precision of our motor capabilities. For instance, our targeting precision for a reaching task without visual feedback is only in the percent range. However, with visual and tactile feedback information available, we are able to adapt our movements in very many ways to achieve an impressive range of sophisticated goals. This provides strong evidence for the view that precision and planning is of secondary importance for flexible movement control; what really counts is the capability to exploit a rich sensory context to shape an only coarsely pre-planned movement continuously towards its goal.

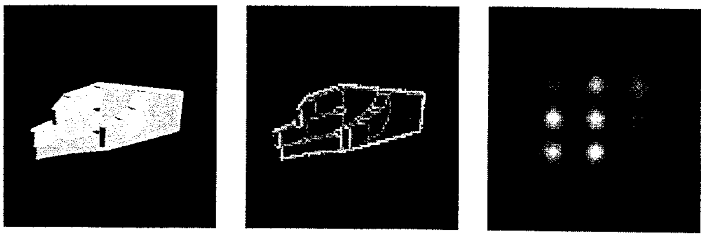

Fig. 2 Preprocessing sequence to obtain input vector from pixel image. Left (a): input image, center (b): edge-image obtained after Laplace filtering and logarithmic intensity transformation, right (c): arrangement of Gaussian kernels used to derive 9-dimensional input vector.

To explicitly program the use of such context, however, can be a very difficult and tedious task. However, if we adopt the approach outlined above, much of this task could be achieved by learning. In the vast majority of cases, the relation between sensor signals and required movements can be expected to be smooth and, therefore, can be represented by a continuous mapping between the quantities representing these data. Usually, a major obstacle for the determination of such mappings is the high dimensionality of the spaces that are involved. Working in low-dimensional representation spaces we can construct good approximations to these mappings on the basis of only a limited number of training examples. This can be achieved efficiently by neural networks which lend themselves excellently as flexible "function approximators". In the next section, we will describe a particular network type, which in addition to learning a smooth mapping, also can optimize the choice of the prototypes used to obtain a low-dimensional description of the input and output data, respectively, and which is therefore a very promising candidate for our approach.

3. Learning with Locally Linear Neural Maps

In this section, we want to illustrate the concepts of the previous section with some concrete simulation results. These concern the task of extracting the three-dimensional configuration of a simulated robot hand from perspective pixel images. The traditional approach would require a sequence of at least several processing steps, such as filtering, edge-detection, segmentation, part identification and finally fitting of a model of the hand shape to the segmented parts thus obtained (see, e.g. Horn 1986). In the neural network approach, only some limited form of preprocessing is necessary. The network can then learn from a set of examples to extract the correct hand postures directly from the preprocessed image data.

We use a network with a single internal layer of units, labelled by an index r. Each unit receives the same input, which was chosen as a 9-dimensional vector x. This vector is obtained from a (computer-generated) image of the hand, such as Fig. 1, by some rudimentary preprocessing (Fig. 2). Fig. 2a shows a typical input image. Applying a 3×3-Laplace mask and clipping any negative values, we obtain an image in which mainly edge-information is preserved. A subsequent logarithmic transformation compresses the dynamic range of the intensity values so obtained. The resulting image (shown in Fig. 2b) is convolved with 9 Gaussians located at the lattice positions indicated in Fig. 2c (to better depict the locations of the Gaussians in Fig. 2c, the displayed widths are only 20% of the widths actually used), and the resulting 9 real values are used as the components x_i of the 9-dimensional input vector x. The task of the network is to learn a mapping from these 9-dimensional representation vectors x to the "mixture coefficients" y = (y_1, y_2, y_3) that determine the contribution of each posture prototype to the hand configuration shown in the original image. Weighting the joint angles of the 3 basis hand postures with these coefficients then yields the hand posture that is "perceived by the network".

The network itself consisted of $N = 20$ units. Each unit r implements a locally valid linear mapping, specified by an output weight vector $w_r^{(out)} \in \mathbb{R}^3$ and a 3 × 9-matrix A_r. In addition, each unit carries an input weight vector $\mathbf{w}_r^{(in)} \in \mathbb{R}^9$. The output $\mathbf{y}_r$ of a unit is given by

$$\mathbf{y}_r = \mathbf{w}_r^{(out)} + \mathbf{A}_r(\mathbf{x} - \mathbf{w}_r^{(in)}). \tag{1}$$

This represents a linear mapping with Jacobian A_r, passing through the point $(\mathbf{x}, \mathbf{y}) = (\mathbf{w}_r^{(in)}, \mathbf{w}_r^{(out)})$. Which of these mappings is used to obtain the output $\mathbf{y}^{(net)}$ of the network is determined by the distances $d_r = \| \mathbf{x} - \mathbf{w}_r^{(in)} \|$. In the simplest case, the unit s for which $d_s = \min_r d_r$ is used ("winner-take-all"-network); usually a somewhat better accuracy can be obtained if a weighted superposition of the contributions of several units is used, e.g. according to

$$\mathbf{y}_{(net)} = \sum_r \mathbf{y}_r f_r, \tag{2}$$

$$f_r = Z^{-1} \exp(-d_r^2/\sigma_r^2), \tag{3}$$

$$Z = \sum_r \exp(-d_r^2/\sigma_r^2), \tag{4}$$

where σ_r is a measure of the radius of the "receptive field" of unit r and may be set, e.g., to $\min_s \| \mathbf{w}_r^{(in)} - \mathbf{w}_s^{(in)} \|$ (Saha and Keeler 1990). The ansatz (2)–(4) for fixed vectors $\mathbf{y}_r$ is also known as generalized radial basis function-approach ("GRBF", see, e.g. Girosi and Poggio 1990) and related to self-organizing maps (Kohonen 1984), which impose some additional structure by generalizing the weights associated with each unit among some subset of "topological neighbors". Note, however, that in contrast to the conventional GRBF-ansatz in our case the vectors $\mathbf{y}_r$ are not constant but instead are linear functions of the input vectors $\mathbf{x}$ that are given by (1). Due to either the winner-take-all-rule (which emerges as the special limiting case $\sigma_r \to 0^+$) or as a result of the exponentials, each of these linear maps contributes only in the vicinity of the respective center $\mathbf{w}_r^{(in)}$. The whole, usually highly non-linear mapping is, therefore, represented as a weighted superposition of many locally valid linear maps instead of as a superposition of a corresponding number of fixed output values. This provides a significantly higher accuracy (Martinetz 1990, Ritter et al. 1991).

Training of the network can proceed in a supervised manner, using a training set of correct input-output pairs $(\mathbf{x}^{(\alpha)}, \mathbf{y}^{(\alpha)})$, $(\alpha) = 1, 2 \ldots M$. Both, input and output weights may be adjusted according to simple error-correction type rules, i.e. no backpropagation is necessary:

$$\Delta \mathbf{w}_r^{(in)} = \varepsilon_1 (\mathbf{x}^{(\alpha)} - \mathbf{w}_r^{(in)}) f_r, \tag{5}$$

$$\Delta \mathbf{w}_r^{(out)} = \varepsilon_2 (\mathbf{y} - \mathbf{w}_r^{(out)}) f_r, \tag{6}$$

$$\Delta \mathbf{A}_r = \varepsilon_3 (\mathbf{y} - \mathbf{y}^{(net)}) (\mathbf{x}^{(\alpha)} - \mathbf{w}_r^{(in)})^T f_r / d_r^2, \tag{7}$$

where the $\varepsilon_i > 0$ are learning step size parameters (a more detailed discussion of these learning rules in the context of self-organizing maps can be found, e.g. in Ritter et al. 1991).

Figs. 3a–c provide some impression of the performance of the network after training ($N = 20$ units, several repeated learning cycles through a data base of $M = 2000$ learning samples). Each picture in the top row shows an input image (not taken from the training set), while the corresponding picture below shows the hand posture as reconstructed by the network. As can be seen, these reconstructions are not entirely accurate; however, they provide a very good account of the correct hand posture. During training, some moderate variation of the viewpoint (rotations and translations) was introduced for each new example. As a result, the network can also correctly identify moderately rotated or translated versions of a posture, providing a useful degree of insensitivity against imprecise centering or

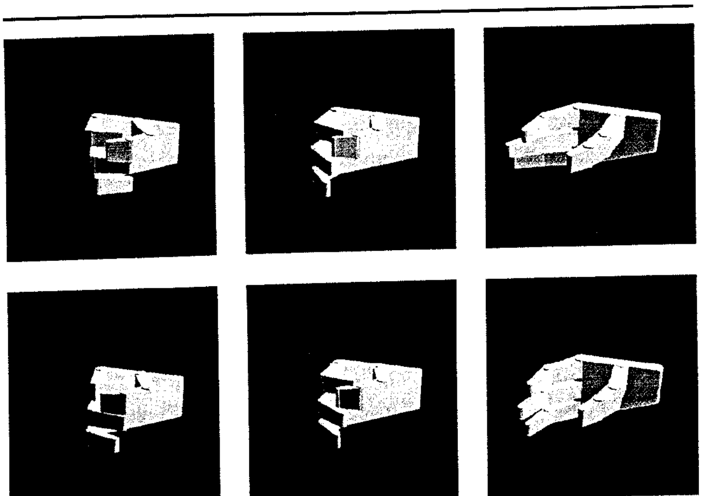

Fig. 3 Performance of the network on some hand postures. Top row: input images. Bottom row: Corresponding images of the 3d-postures reconstructed by the network on the basis of the 9-d-input vectors obtained by the preprocessing steps outlined in Fig. 2.

moderate changes of the view-direction of the scene.

The present level of performance has been achieved without any sophisticated optimizations. There is still ample room for improvements: better preprocessing schemes can be devised to improve invariance, mechanisms for focusing attention to a subfield of the image may make particularly precise extraction of local posture information possible and several subnetworks may cooperate in a parallel or hierarchical fashion to assemble partial posture information into more global representations. Future research will adress these and related issues to explore the potential of neural approaches to sensory-motor control in the important domain of controlling visually guided grasp movements.

4. Movement Coordination in Biological Systems

Observation of the leg movements of a walking animal in the field shows that the legs are well coordinated in a specific gait. This spatio-temporal movement pattern of the legs seems to be quite fixed and its details only vary with walking speed. The pattern is in fact extremely stable with respect to disturbances which may result from an unpredictable environment as for example uneven surfaces. In some cases the underlying control system even copes with the problem of the loss of a leg. How is a system organized which is responsible for the coordination of leg movement? This is another example where an algorithmic solution is possible but requires much more time for computation if reactions to all sorts of disturbances are to be taken into account.

Biological experiments have been performed with six-legged insects or 8-legged crustaceans. On the basis of these experiments we know that this coordination is not achieved by means of a central controller. Rather, each leg has its separate sensory-neural unit which controls the movement of the leg. Each unit can be influenced by neighbouring units by way of specific signals enabling coordination of the movement of the different legs (for a review, see Cruse 1990). The coordination pattern can be regarded as an emergent property resulting from the local couplings between the units. Whereas for the stick insect six types of coordinating signals have been found, the crayfish seems to be much more simply organized as only three types of influences are sufficient to describe its behaviour. To give an example, Fig. 4 schematically presents those two types which act between ipsilateral legs, i.e., between neighbouring legs of the same side of the body. The two traces represent the movement of the legs. These consist of two parts, the power stroke (downward deflection) and the return stroke (upward deflection). During the power stroke the leg is moved to the rear and supports the body; during the return stroke the leg is swung back to the initial position to start the next power stroke. In Fig. 4 one leg is plotted in different phase situations relative to the other leg. These phases might have been produced by disturbances of the normal walk. One coordinating mechanism is only rostrally oriented, i.e., acts only from the posterior (rear) to the anterior (front) leg. This is illustrated in Fig. 4A: as long as the posterior leg performs a power stroke, the anterior leg has to perform or continue a return stroke. In addition, the velocity of the movement during the return stroke is also decreased to some extent. Thus the return stroke can be prolonged so that normal coordination is regained in the next step. The vertically striped bar indicates the time during which this influence is active. The intensity of the influence is roughly indicated by the thickness of the bar. The second

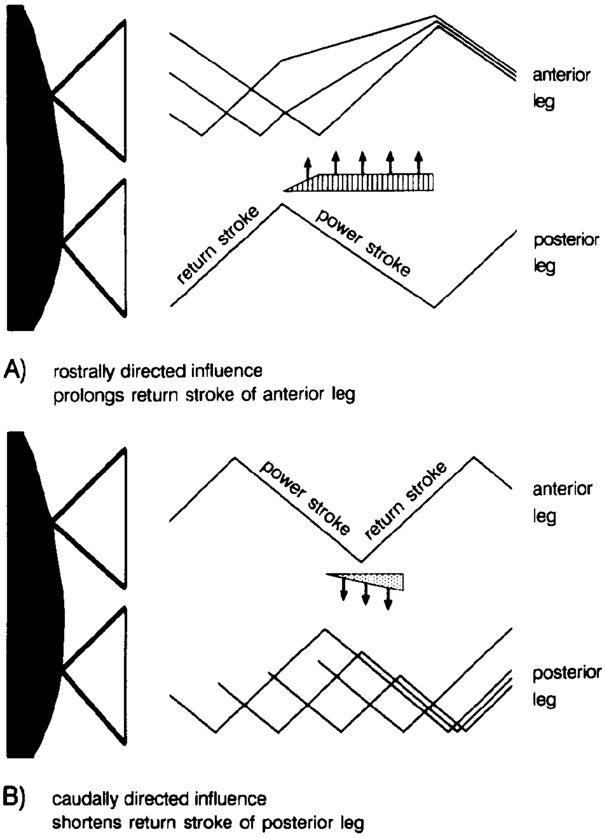

Fig. 4 Coordination between the ipsilateral legs of a crayfish. The upper traces show the anterior leg. Abscissa is time, ordinate is position of the leg tip in a body fixed coordinate system. Each schema is drawn as if only one of the two coordinating mechanisms existed. In each case the influencing leg is drawn only once. For the influenced leg several traces are presented to show the effect of the coordinating mechanism. The duration and intensity of the influences are roughly indicated by the length and thickness of the wedges, respectively. (A) The rostrally directed influence is active during the power stroke of the posterior leg. It prolongs the return stroke of the anterior leg and can also decrease the speed of the limb movement. (B) The caudally directed excitatory influence is active at the end of the power stroke and the beginning of the return stroke of the anterior eg. It "excites" the start of a power stroke in the controlled, posterior leg.

influence, illustrated in Fig. 4B, is caudally directed: when the anterior leg is near the end of its power stroke or at the beginning of its return stroke, an influence with increasing intensity has the effect of ending the return stroke and starting the power stroke of the posterior leg, thus shortening the return stroke of the latter. Again normal coordination is regained within one step. To couple contralateral legs, i.e., legs of opposite sides of the body, the crayfish uses a mechanism that closely resembles the ipsilateral, caudally directed influence. In contrast to ipsilateral influences, the contralateral ones act in both directions between the two legs.

The normal gait of the animal results from the fact that the units of all neighbouring legs are coupled by these local rules, as described above. Since, therefore, this system is found to be of an inherently parallel nature, it is well suited to be simulated by means of a neural network. The sensory-neural units, which have been described as controlling the movement of a leg, do themselves not correspond to single neurons but represent a more complicated system. For the sake of clarity we will simplify the system and consider only one property of the leg namely the performing of more or less rhythmic forward-backward movements, the alternating return and power strokes. This unit can then be considered as a simple oscillator. The oscillator includes sensory feedback because the transition from power to return stroke and vice versa is influenced not only by signals from the other legs, but also by sensory signals from its own leg. Thus each oscillator consists of several neurons. Our first aim is to build a network which consists of coupled neuronal oscillators. The whole system should produce a properly coordinated leg pattern which can compensate for external disturbances as fast as the animals can. To begin with we plan to utilize our knowledge obtained from biological experiments as much as possible for the structure of the network. Improvement by means of learning algorithms should only be used if the behaviour of this model shows deficiences.

As mentioned, the simplified model will only solve the problem of producing coordination between the legs. The question of how the different joints of the individual leg are coordinated to produce an appropriate leg movement during the power or the return stroke was not considered. This problem is particularly interesting when the leg becomes redundant, i.e., when it has more degrees of freedom than necessary for the task at hand. We investigated this question using as an example the control of the human arm and found that the behaviour could be explained by assuming the application of four rules (Cruse and Brüwer 1987). These are, first, an equal contribution of all joints to the movement (this corresponds to the well-known pseudo-inverse control); second, the minimization of the static costs by means of a cost function applied to each joint; third, the minimization of the inertial forces acting at the tip of the end effector by following a straight line in the workspace; and forth, by avoiding movements which are strongly non-monotonic in the joint space (this probably decreases dynamic costs).

These rules can be described by an algorithmic approach but the above-mentioned advantages of neural network systems ask for a simulation by means of these principles. Using a simple 3-layer feedforward network we successfully simulated one example of these rules (no. 2) (Brüwer and Cruse 1990). However, control of the actual leg or arm requires the use of sensory information. Therefore a network containing feedback channels is appropriate and will therefore be investigated in our project. This system, controlling a multilimbed leg, will then be implemented in the abovementioned oscillator.

One particular problem remains to be solved in relation to the movement during power stroke. Walking on uneven surfaces means that the body-to-ground distance has to be adapted individually for each leg. For this case, too, information is available from biology: in the stick insect the vertical distance of each leg ("height") is subject to a proportional position controller (Cruse 1976). As the rearward movement component is subject to velocity control two different feedback systems control the same final elements. Thus we have a case of shared control which has to be solved by the network. As these controllers contain dynamic properties, the network has to cope with time derivatives and properties of temporal filters. In this case we also start designing the network by introducing as much a priori knowledge as possible; but later the question of how to learn dynamical properties will also be approached.

As mentioned, we are studying the network which controls the movement of a multilimbed leg. This "leg" can, of course, also be an arm or a robot manipulator. In this context it is of interest to know how the path of the end effector is planned. This is particularly interesting if the workspace contains obstacles. On the basis of experiments with human subjects we are currently investigating the strategies of human beings and plan to implement these strategies in a neural network model for path planning. These models could then be used to control the movement of a robot manipulator or to plan the path of a freely moving autonomous robot.

5. Conclusion

One of the aims of the research reported in the previous sections is to circumvent the knowledge acquisition bottleneck currently burdening the desing and the construction of more intelligent and flexible robots. Neural networks seem to offer a framework that is particularly promising for approaching this goal. Several reasons can be given in support of this expectation: first, neural networks are well suited to implement learning algorithms, particularly in the sensory-motor domain, where continuous and noisy signals need to be processed, and where traditional, symbol-oriented approaches and purely analytical methods encounter great difficulties. The results reported in Section 3. provide a concrete example, showing that a visual recognition task, which, if to be achieved by conventional image processing techniques, would require a fairly sophisticated multi-stage system capable of identifying a complex object of variable shape, can be achieved by a rather small neural network on the basis of a modest number of training examples. By systematically exploring which kinds of input representations and network architectures are particularly useful for specific types of tasks, we may expect to gradually gain the capability to build larger and significantly more competent systems, for which a large portion of knowledge acquisition may boil down to the presentation of examples. An important role on this way will be played by biology: by taking information gained from biological organisms seriously into account for the design of artificial neural systems, we may be able to develop technological approaches that would be extremely hard – or, maybe even impossible – to arrive at by theorizing alone. From this point of view, neural network models emerge as a natural tool to efficiently integrate knowledge from basic biological research with existing approaches. The work reported in Sec. 4 provides a concrete example of this second way of approach. Here, control strategies for the coordination of complex multi-limb movements that are robust against a wide range of disturbances are adopted from nature and implemented in artificial networks that then make these strategies usable for robust and yet computationally affordable real time control of artificial walking machines.

Acknowledgement

This work has been supported by the Bundesministerium für Forschung und Technologie (grant numbers ITN 9104A0 and ITN 9104B2). Any responsibility for the content of this publication is with the authors.

References

Arbib M. A., Amari S. I, (1985): Sensori-Motor Transformations in the Brain. J. theor. Biol. 112: 123–155

Brady M. (1990): Robotics Science. MIT-Press

Brooks R. (1990): Elephants Don't Play Chess. Robotics and Autonomous Systems 6: 3–15

Brüwer M., Cruse H. (1990): A network model for the control of the movement of a redundant manipulator. Biol. Cybern. 62: 549–555

Cruse H. (1976): The control of body position in the stick insect (Cavansius morosus) when walking over unveven surfaces. Biol. Cybern. 24: 25–33

Cruse H. (1990): What mechanisms coordinate leg movement in walking arthropods? Trends in Neurosciences 13: 15–21

Cruse H., Brüwer M. (1987): The human arm as a redundant manipulator: The control of path and joint angles. Biol. Cybern. 57: 137–144

Girosi F., Poggi T. (1990): Biol. Cybernetics 63: 169–176

Grossberg S., Kuperstein M. (1986): Neural Dynamics of Adaptive Sensory-Motor Control. North Holland, Amsterdam

Hertz J., Krogh A., Palmer R. G. (1991): Introduction to the Theory of Neural Computation. Addison-Wesley, Redwood City

Hildreth E. C., Hollerbach J. M. (1985): The Computational Approach to Vision and Motor Control. AI-Memo 846, Massachusets Inst. of Technology

Horn (1986): Robot Vision. McGraw-Hill, New York

Kawato M, Furukawa K, Suzuki R (1987): A Hierarchical Neural-Network Model for Control and Learning of Voluntary Movement. Biol Cybern 57: 169–185

Kohonen T. (1984): Self-Organization and Associative Memory. Springer Series in Information Sciences 8, Heidelberg

Kuperstein M. (1987): Adaptive Visual-Motor Coordination in Multijoint Robots using Parallel Architecture. Proc IEEE Int Conf Automat Robotics 1595–1602, Raleigh NC

Martinetz T., Ritter H., Schulten K. (1990): Three-dimensional Neural Net for Learning Visuomotor-Coordination of a Robot Arm. IEEE-Transactions on Neural Networks, Vol. 1, No. 1, pp 131–136

Ritter H., Martinetz T., Schulten K. (1991): Neuronale Netze. Addison-Wesley Bonn, 2. erw. Auflage

Rumelhart D. E., McClelland J. L. (1984): Parallel Distributed Processing, Cambridge Massachusets: MIT-Press

Saha A., Keeler J. D. (1990): Algorithms for Better Representation and Faster Learning in Radial Basis Function Networks. In: Advances in Neural Information Processing Systems 2 (D. S. Touretzky ed.) 482–489 (Morgan Kaufman, San Mateo)

Berichtigung

In Heft 4/91 der **it** wurden auf Seite 198 versehentlich die Namen der Autoren ihren Lichtbildern falsch zugeordnet.

Hier die korrekte Namenszuordnung in der Reihenfolge der Bilder von oben nach unten:

Prof. Dr.-Ing. Rolf Hoffmann,
Prof. Dr.-Ing. Hans Liebig,
Dr.-Ing. Thomas Flik