Obstacle avoidance in a model of human reaching behavior.

OBSTACLE AVOIDANCE IN A MODEL OF HUMAN REACHING BEHAVIOR

A Thesis Presented

by

LOUKIA D. LOUKOPOULOS

Submitted to the Graduate School of the
University of Massachusetts Amherst in partial fulfillment
of the requirements for the degree of

MASTER OF SCIENCE

May 1994

Department of Psychology

OBSTACLE AVOIDANCE IN A MODEL OF HUMAN REACHING BEHAVIOR

A Thesis Presented

by

LOUKIA D. LOUKOPOULOS

Approved as to style and content by:

David A. Rosenbaum, Chair

Neil Berthier, Member

Arnold Well, Member

Charles E. Clifton Jr., Department Head
Department of Psychology

ACKNOWLEDGMENTS

For their valuable feedback and unlimited source of ideas, I thank Jonathan Vaughan, Ruud Meulenbroek, Sascha Engelbrecht, Martin Fischer, and all the members of my Committee. For his unsurpassed support and inspiration, I thank my advisor, David Rosenbaum. My gratitude, finally, extends to my family and all those who have been by my side throughout my studies.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

REACHING

Movement is, without doubt, one of the most important functions of all animals. It is what has shaped the evolution of all species by allowing for adaptation and survival on a heterogeneous and densely populated planet. Movement serves a multitude of vital functions: exploration and manipulation of the environment, contact with other beings and objects, feeding, and mating, to name but a few. While it is a behavior we largely take for granted, it is a highly complex function involving the coordinated and skillful manipulation of intricate but delicate bodies.

Given the complexity of movement, students of this behavior have traditionally chosen to examine simplified, goal-directed acts requiring motion. One task that has been studied extensively is the act of reaching, or bringing some part of the body into contact with a target. Reaching is an interesting topic of study largely due to the variety of behaviors it entails. The evolution of reaching can be traced with age, allowing one to make inferences about its role in development. Its reliance on visual perception can reveal interesting facts about both the motor and the perceptual system. Finally, reaching offers the grounds on which to study the ideas of mental representations and motor programs, as well as issues of intra- and inter-limb coordination and movement generation in general. The study of reaching, however, is by no means simple.

The Degrees of Freedom Problem

Consider the task of bringing the hand to a specified point in space. The arm's configuration in three dimensions is controlled by seven variables. These variables, or degrees of freedom (*DOF*), are the seven ways in which the shoulder, elbow, and wrist joints can independently move the arm segments. The shoulder allows for flexion/extension, abduction/adduction, and lateral/medial rotation of the arm. It therefore has 3 *DOF*. The elbow allows for flexion/extension and pronation/supination of the lower arm (2 *DOF*), and the wrist allows for flexion/extension and abduction/adduction of the hand (2 *DOF*). A point in space, however, can be accurately defined by three variables: its x, y, and z positions in Cartesian space. In principle, therefore, only three degrees of freedom are strictly necessary to bring the arm end-effector to any point in space.

It becomes evident from this example that an asymmetry of number of degrees of freedom exists between task and means specification. A hand location in extrinsic space, with the exception of locations at the edge of the work-space, can be achieved with more than one set of arm joint angles. The selection of the appropriate arm configuration, therefore, is ill-posed and is known as the *inverse kinematics* problem. The latter, in turn, is an instantiation of a feature prevalent in many other aspects of human behavior known as the *degrees of freedom* problem (Bernstein, 1967). In general, the degrees of freedom problem arises when more than one solution exists for a given problem.

Studying and Modeling Reaching

The study of reaching or pointing in the laboratory is not easy. Due to the excessive number of degrees of freedom in the human body (circa 100, Turvey, 1990a), researchers have traditionally resorted to investigating the kinematic and/or dynamic characteristics of motion under various constraints. In such studies, great care is taken to ensure movement only by the joints under investigation, and only in the spatial dimensions of interest (e.g., horizontal plane movements: Abend, Bizzi, & Morasso, 1982; Cruse, Brüwer, & Dean, 1993; sagittal plane movements: Atkeson & Hollerbach, 1985; Fischer, Rosenbaum, Loukopoulos, & Szymkowiak, 1993). The results have prompted investigators to model various aspects of reaching behavior along three major lines: muscle equilibrium-point control (Asatryan & Feldman, 1965; Bizzi, 1980; Kelso & Holt, 1980), synergies (Bernstein, 1967; Turvey, 1990b), or cost functions (Flash & Hogan, 1985; Uno, Kawato, & Suzuki, 1989; Brüwer & Dean, 1993) (for a review, see Rosenbaum & Krist, in press). Most approaches, however, have selectively focused either on the problem of movement selection or that of movement generation.

The work presented here is based on the belief that a global approach to the degrees of freedom problem is necessary to study reaching effectively. Various cognitive, neural, and physical aspects of the human body and mind, therefore, should be taken into consideration. A recent attempt to incorporate such concerns into a model of movement selection and generation is the *Knowledge Model* (Rosenbaum, Engelbrecht, Bushe & Loukopoulos, 1993a, b; Rosenbaum, Loukopoulos, Meulenbroek, Vaughan, & Engelbrecht, 1993c).

While the model succeeds in fulfilling the goals it was initially set out to attain, it also has certain limitations. One such limitation, which I address here, is the model's inability to explain reaching around obstacles.

CHAPTER 2

THE KNOWLEDGE MODEL

The Computational Approach

In this thesis I present work that I have done over the past few years on a computational theory of human reaching behavior, the Knowledge Model (Rosenbaum et al., 1993a, b; Rosenbaum et al., 1993c). This model is designed to explain and predict movement planning and generation in the context of simple reaching tasks. It accounts for a variety of phenomena observed in human motor behavior, but is still in need of elaboration and extension. The work presented here focuses on ways in which the theory may be extended.

The core of my work is computational. There are a number of reasons why I feel this is an excellent way to study a complex problem such as reaching. In general, the development of a computational model requires the bottom-up construction of a theory based on ideas derived from experimental findings. These ideas, which should be explicitly formulated, must be incorporated into plausible algorithms. Possible outcomes and/or problems of suggested solutions have to be anticipated and thoroughly tested.

At the same time, a computational model lends itself to simulation. In an area of study such as reaching, where the apparent "naturalness" of movements is important, simulations are a good tool to inspect a theory's predictions. Simulations are also amenable to comprehensible and persuasive real-time presentation, while they allow for fast testing of alternative ideas. A computational model, finally, can be directly fit to experimental data, thus allowing for its evaluation. A thorough evaluation, in turn, may reveal possible weaknesses, suggest possible modifications, or generate ideas for experiments.

Overview

The Knowledge Model is a computational theory aimed at explaining and predicting movement planning and generation in the context of simple reaching tasks. Specifically, it solves the problem of selecting appropriate end-state postures (the inverse kinematics problem), and of executing the postural transition between starting and ending states. It is not, however, restricted to the domain of motor control. In fact, it can be viewed as a general theory of any cognitive function characterized by the degrees of freedom problem (e.g., the proper positioning of articulators for speech production, or the recovery of an object's 3-dimensionality from its projection on the 2-dimensional retina).

At present, the theory addresses the act of bringing some part of the body, termed the <u>contact point</u>, to a specified spatial target. The problem inherent in this situation is the following: every joint on a person's body can be thought of as a degree of freedom[1]. Numerous combinations of angles for each of these degrees of freedom yield body postures which may bring the contact point to the same target location. Figure 1 depicts one such case. In Panel A, the actor brings the hand to a spatial location (denoted by a circle) at the edge of the work-space. This can only be achieved in one way, given a fixed seated position: by flexing the torso as far as possible and by extending the upper and lower arm as far as possible. In the subsequent panels, however, the actor can reach for another, common spatial location by employing at least three different postures or combinations of joint angles. The Knowledge model was primarily designed to address the question of how people select a particular posture in view of the large number of available alternatives.

This model has been implemented as a computer simulation for better presentation of the theory's predictions and implications. The simulation involves the animation of a seated stickfigure viewed from the side which, with 3 degrees of freedom (motion around the hip, shoulder, and elbow), can reach for spatial locations in a 2-dimensional work-space. Movement is carried out in a sagittal plane that contains the stickfigure's shoulder. All examples and figures from here on will refer to this situation. It is important, however, to keep in mind that all algorithms are fully extendible to 3-dimensional space and to more than just 3 *DOF*. In addition, while not specifically addressing neurophysiological issues, the Knowledge Model is designed to be biologically plausible and draws on findings reported in the literature.

For reasons of simplicity, the theory explains reaching in purely kinematic terms. That is, it does not take into account any effects due to interactional forces between limbs (i.e., Coriolis forces, reaction forces, and centripetal forces), or gravity and balance. It is noteworthy, however, that the model is characterized by great success even without having taken such factors into account. Efforts are being made to incorporate the effects of dynamic factors into the existing version of the Knowledge Model.

There are three major components comprising this theory: Posture Planning, Movement Execution, and Learning. First, I present an outline of the major assumptions upon which the Knowledge Model rests, followed by a brief description of each component. Because the focus of this thesis is not primarily on the theory itself, the interested reader is referred to the theory's previous expositions (Rosenbaum et al., 1993a, b; Rosenbaum et al., 1993c).

Assumptions

Knowledge of Postures

The major tenet of the Knowledge Model is its reliance on body postures. A posture, P_i, can be thought of as a vector defined by the set of n numerical values, one for each of the degrees of freedom on the human body (i.e., in the case of the simulation, n equals 3). Each value denotes the angle of rotation of a limb with respect to its adjoining limb, within a rotational plane. The convention adopted is to work from the ground up, so that the hip angle is defined as the torso angle with respect to the upper leg, the shoulder angle is defined as the upper arm angle with respect to the torso, and so on. The actor is assumed to possess a knowledge-base of postures ($\mathbf{P} = \{P_1, P_2, ..., P_i, ..., P_m\}$). Although the contents of the knowledge-base are free to vary, the number of elements it can contain is fixed. Note, however, that these postures are not indexed by the extrinsic contact point locations in which they result.

The notion of storing postures is supported from literature showing that people are, in fact, more sensitive to starting and ending states than to movements. The mass-spring model of movement (Feldman, 1966; Kelso & Holt, 1980; Bizzi, Hogan, Mussa-Ivaldi, & Giszter, 1992), an influential model of motor control, suggested that limb positions are specified by adjusting the lengths of muscles acting on a joint so an equilibrium between opposing muscle forces is achieved. Muscle lengths, however, imply joint angles (Shadmehr, 1993), or in other words postures. Research on the psychophysical perception of movement comfort also seems to support the notion that the motor system "cares more" for the comfort achieved at the end of a reaching task rather than at the beginning or the duration of the movement (Rosenbaum, Marchak, Barnes, Vaughan, Slotta, & Jorgensen, 1990; Rosenbaum, Vaughan, Jorgensen, Barnes, & Stewart, 1993d). Overall, the dimensionality of posture-space is clear (i.e., joint angles), and storing postures may be more economical for memory allocation and the best of a number of costly alternatives (e.g., storing entire trajectories or muscle length-tension functions).

Postures are assumed to be initially acquired and stored during development by acts of random reaching around one's work-space. In the case of the simulation, they are randomly generated except for being constrained to fall within the humanly possible motion ranges of each joint or degree of freedom. The process by which the contents of this knowledge-base may be altered is described later in this chapter, in the section labeled Learning.

Forward Kinematics

A second assumption of the Knowledge Model is that the actor, given a particular posture, has the ability to compute where any given point on his/her body is in extrinsic space. This computation, called *forward kinematics*, is a relatively simple trigonometric procedure and makes use of the following equations:

$$x_j = x_{j-1} + (\ell_{j-1} \cos \sum_{j=1}^{n-1} \theta_j) \quad \text{and} \tag{1}$$

$$y_j = y_{j-1} + (\ell_{j-1} \sin \sum_{j=1}^{n-1} \theta_j), \tag{2}$$

where x_j and y_j are the horizontal and vertical positions of joint j in Cartesian space, ℓ_j is the length of the limb whose origin is at joint j (e.g., the torso's origin is at the hip joint), and θ_j is the degree of counterclockwise rotation between the extension of limb ℓ_j and limb ℓ_{j+1}. It is assumed that the spatial location of the first joint, as well as the lengths of all limbs are known. Figure 2 shows a 3 *DOF* stickfigure along with the conventions used to illustrate the notation above.

It is postulated that the actor computes forward kinematics on the spot, as the need arises. To understand why, consider the following: if a stored posture were to be indexed by its resulting body point extrinsic location, which body point's location should be computed and stored? A possible answer would be to store the locations of all points on the human body. This is obviously problematic given the infinite number of body points. Another possible answer would be to store only joints' extrinsic locations, but that may again be too expensive given the number of joints on the human body. A third alternative would be to store the location of a single body point (e.g., that of the index finger). This solution, however, begs the question of how we are able to reach, when necessary, with other parts of the body, or even with hand-held tools. The most viable solution, therefore, seems to be to compute forward kinematics only after the contact point has been specified. In such a manner, the required resources for information storage are minimized, while reaching or pointing can still be achieved with any part of the body.

Model Components

Posture Planning

Consider the task of reaching for a specified location in extrinsic space (the target location) with the hand, given a starting body posture. The first problem is to select a single posture out of the numerous possible postures which could bring the hand to the target. The goal of the first component of the Knowledge Model is to compute this target posture, $\bar{P}$, which brings the end-effector into contact with the desired target location. To achieve this, all stored postures are assumed to place "bids" with respect to their suitability in completing this task on their own. Suitability is determined on the basis of spatial accuracy and energy consumption. The sum of all posture vectors, weighted by their respective bids, yields the target posture.

Spatial Error Cost

Initially, all stored postures are evaluated with respect to their spatial accuracy. Each posture P_i, is assigned a spatial error cost, $D(P_i)$, corresponding to the Euclidean distance between target location (x_t, y_t) and the contact point location (x_c, y_c) if posture P_i were to be adopted:

$$D(P_i) = \sqrt{(x_t - x_c)^2 + (y_t - y_c)^2}. \tag{3}$$

Note that this computation requires forward kinematics in order to find the resulting contact point location given posture P_i since, as mentioned before, stored postures are not indexed by extrinsic locations. At this stage, however, this computation can be carried out because the contact point for the task in question has been specified.

Travel Cost

Each stored posture P_i is also assigned a travel cost, $V(P_i)$, representing the energy that would be expended if the actor were to adopt that posture. In effect, this cost addresses another question — that of how costly it would be to move from the starting posture to the stored posture under evaluation. To compute the travel cost, three factors must be known about each joint comprising the stored posture.

The first factor is the mobility of each joint, which we label the expense factor, k_j. This free parameter is influenced by characteristics of the modeled system, such as the joint stiffness, the friction among joints, and the moments of inertia of the limbs. Given that joints control the motion of different segments of the body, it is reasonable to assume that they have different expense factors. The hip joint, which controls movement of the torso, for example, presumably has a larger expense factor than the shoulder joint, which controls movement of the lighter upper arm. The expense values, however, are not fixed; they may change through experience or injury.

The second factor is the angular displacement, α_j, each joint has to undergo. This is simply the absolute difference between the angle of joint *j* at the starting posture and the angle of joint *j* at the stored posture under evaluation.

The third and last factor is the movement time, MT_j, allotted to each joint for the completion of its required angular displacement. The movement time can either be an externally specified time T_j (e.g., as dictated by an experimenter), or generated by the model itself. The latter is achieved by relating each joint's required angular displacement, α_j, to its expense factor, in a fashion reminiscent of an amplitude-frequency optimum curve (Rosenbaum, Slotta, Vaughan, & Plamondon, 1991). In this manner, it is possible to compute each joint's preferred movement time, T_j^*. This is the time in which it would be optimal to cover a required amplitude α_j with joint *j*:

$$T_j^*(\alpha_j) = k_j \ln(\alpha_j + 1). \qquad k_j \geq 0 \qquad (4)$$

Adding 1 to the angular displacement prevents $T_j^*(\alpha_j)$ from becoming undefined when α_j is any non-negative number less than 1.0.

Once these three factors have been specified, the cost $V_j(\alpha_j, T_j)$ of moving joint *j* through an angular displacement α_j in some time T_j is found by:

$$V_j(\alpha_j, T_j) = \left(\frac{k_j \alpha_j}{r}\right) \left(1 + \frac{(MT_j - T_j^*(\alpha_j))^2}{s^2}\right), \qquad (5)$$

where *r* denotes the unit of angular displacement (1 degree), and *s* denotes the unit of time (1 millisecond). Both terms are introduced to make the expression dimensionless. Once again, MT_j is an externally specified movement time, or the preferred time as computed from Equation 4, and T_j^* is the preferred movement time.

The travel cost, $V(P_i)$, of moving from the starting posture to posture P_i, finally, is computed by simply adding the travel costs of all joints participating in the movement from the starting posture to the stored posture under evaluation:

$$V(P_i) = \sum_{j=1}^{n} (V_j(\alpha_j, MT_j). \qquad (6)$$

Published data have not been very indicative of the timing aspects involved in coordinated joint movements. In the simulations considered here, I have adopted the assumption that all joints start and end moving together and so it is necessary to define a common movement time. The idea behind this is to find the common <u>optimal movement time</u>, T*, such that the travel cost in traveling between the starting posture and the stored posture under evaluation is minimized. To compute that time we take the weighted average of all joints' preferred movement times:

$$T^* = \frac{\sum_{j=1}^{n} k_j \alpha_j T_j^*(\alpha_j)}{\sum_{j=1}^{n} k_j \alpha_j} \, . \qquad (7)$$

Total Cost

Each stored posture now has a spatial error cost and a travel cost computed for a task in question. A weighted sum of the two costs yields the total cost, $C(P_i)$, of each posture:

$$C(P_i) = w_d \left(\frac{D(P_i)}{MaxD}\right) + w_v \left(\frac{V(P_i)}{MaxV}\right) . \qquad (8)$$

The two non-negative weights (spatial error weight, w_d, and travel cost weight, w_v) reflect task demands and always sum up to 1.0. The importance of these two parameters will be discussed further in the section labeled Task Demands. Note that each cost is normalized after it is divided by the maximum value for that same cost (MaxD and MaxV) among all stored postures for the reaching task in question.

Weights

Stored postures now have to be assigned weights that appropriately reflect their suitability for the task in question. The minimum of all stored postures' total costs becomes the standard deviation, σ, of a Gaussian distribution function:

$$G(C(P_i)) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(C(P_i)-\mu)^2}{2\sigma^2}\right) . \qquad (9)$$

By setting μ equal to 0 (the ideal total cost), this calculation yields a posture's Gaussian value, $G(C(P_i))$, given its total cost, $C(P_i)$. In effect, all total costs are positioned on the abscissa of the Gaussian and the corresponding value on the ordinate yields that stored posture's Gaussian value[2]. The lower the total cost, therefore, the larger the Gaussian value of a given posture.

To derive the weight, $g(P_i)$, of posture P_i we take the ratio of that posture's Gaussian value over all m stored postures' Gaussian values:

$$g(P_i) = \frac{G(C(P_i))}{\sum_{i=1}^{m} G(C(P_i))}. \tag{10}$$

The metaphor of stored postures placing bids for a task helps clarify the purpose of using a Gaussian distribution to determine the weights. In effect, it is a noise-attenuating filter which ensures that if there exists in the knowledge-base a posture, P*, which is highly suitable for the task, it should be given a high weight, whereas all other bidders should be silenced. This is achieved because P*'s total cost would be very low, and consequently σ would be very small, yielding a slim and tall Gaussian distribution. Most total costs on the abscissa, therefore, would correspond to values on the ordinate where the Gaussian distribution is approaching zero. If, on the other hand, no posture is highly suitable, all input should be encouraged and all postures should have small but more uniform weights (large σ, fat and short Gaussian distribution).

Target Posture

The final step in the Posture Planning component is the determination of the target posture, $\bar{P}$, achieved by taking the weighted sum of all postures in the knowledge-base[3]. Given the stated constraints, the target posture is the most appropriate for bringing the end-effector to the desired target location and is computed as follows:

$$\bar{P} = \sum_{i=1}^{m} g(P_i)P_i \,. \tag{11}$$

Task Demands

As mentioned above, postures' weights can be set to reflect specific task demands. Given that a posture's weight depends on its total cost, the weights for the two costs (w_d and w_v) may be appropriately manipulated so that a posture is evaluated with respect to the importance of these factors in the particular task. This, in turn, will be reflected in the selected target posture. If, for example, a task requires high accuracy, the spatial error cost weight, w_d, is set high so that all postures with very low spatial error costs, regardless of their travel costs, are assigned small total costs and, in turn, given large weights. If, on the other hand, a task also requires energy economy, only those postures that achieve a good level of accuracy combined with efficiency will be assigned large weights.

To illustrate the role of task demands, Figure 3 shows an extreme situation of a stickfigure with knowledge of only 3 stored postures (A, B, and C). These were evaluated twice for their suitability for reaching to a target (T). The first time (Panel A), the spatial error weight was set equal to 0.79 (hence, the travel cost weight was equal to 0.21). Posture B received the highest weight because it achieved good spatial accuracy — the dimension of importance. Posture A received a lower weight because its accuracy was not quite as good as that of posture B. Still, the weight it received allowed it to make a contribution to the weighted sum. Posture C, on the other hand, received a minimal weight because it failed to bring the hand close to the target. The result of taking the weighted sum of these 3 postures was the target posture (TP) which is an average of postures A and B.

The second time (Panel B), the spatial error weight was lowered to 0.21 and the three postures received different weights than before. Now, posture A received the highest weight based on the good level of spatial accuracy it achieved. This time it was weighed more than before because it was also efficient. Posture B, on the other hand, now had a lower weight because it was not as efficient as posture A — it requires a large angular displacement of the hip and shoulder in the transition from the starting posture (S). Posture C, finally, still failed to fulfill any of the task requirements. The resulting target posture, therefore, was again an average of postures A and B. This time, however, it was more biased towards posture A as indicated by the denoted joint angles.

Feedforward Correction

As stated before, stored postures are assumed to be randomly generated. Figure 4 shows a scatter plot of all hand locations resulting from a given knowledge-base of about 600 postures. Contrary to one's expectation, these are not equally distributed around the work-space. A non-linearity in the mapping of joint angle configurations on spatial end-effector positions exists (Bullock, Grossberg, & Guenther, 1993), because more than one posture can bring the hand to the same spatial location: some areas contain numerous, not to mention overlapping, hand location symbols, whereas other areas contain relatively few symbols. When taking a weighted sum of all stored postures to derive the target posture , therefore, it is possible that the desired spatial accuracy is not met. In other words, the target posture may fail to bring the hand close enough to the target because the target location is in an area which is not as well represented. The final feature of the Posture Planning component is a correction mechanism, *feedforward correction*, designed to address this possibility.

The spatial accuracy demands of a particular task are reflected in an acceptable distance factor, A, whose value is set equal to an externally imposed task constraint, such as a tolerance region around the target point. The units for A can be thought of as body-scaled distance units (i.e., the hand's length). Once a target posture is derived, the signed error between the contact point and the spatial target (in extrinsic coordinates) is computed. If this is found to be larger than A, the target posture computation algorithm is repeated. This time the actor aims for a virtual target some distance away from the original target, in the direction opposite from that in which the error occurred. This distance is a proportion, β, of the distance between the contact point (CP) given the target posture and the original target location (T):

$$B_{d(c)} = B_{d(c-1)} + \beta(T_d - CP_d)_{(c)}. \tag{12}$$

Here, B denotes the correction bias, d denotes the Cartesian (x or y) dimension in which the bias is introduced, β denotes gain ($\beta \geq 0$) and c denotes the number of correction cycles. Our simulations have established that the optimal value for β is approximately equal to 0.5. Feedforward correction is executed after a target posture is computed and is repeated until either the acceptable distance is achieved, or a time limit is reached (a pre-specified number of cycles is completed). In this fashion, the target postures computed after each cycle succeed in bringing the hand increasingly close to the target.

Feedforward correction can occur in two ways: covertly or overtly. In the first fashion, all computations are done mentally and the actor only moves to the target posture computed after the last cycle — the posture assumed to be the most accurate. The advantage of correcting errors in a covert fashion is that it is likely to be more energy- and time-conserving. Small corrective movements, however, are sometimes observed at the end of aiming motions (Woodworth, 1899; Crossman & Goodeve, 1963/1983; Keele, 1968). Such a situation, therefore, can be simulated by simply rendering the results of the feedforward correction mechanism overt. In other words, the actor could be shown moving through all target postures successively computed after each correction cycle until the spatial accuracy is met. The best overall feature of the mechanism, however, is that the number of feedforward cycles provides an empirically testable index of planning time.

Movement Execution

Moving from the starting posture to the computed target posture is, once again, a problem with an infinite number of solutions. One can imagine achieving this postural transition by employing various timing combinations which, in turn, would yield markedly different hand trajectories. For a solution to this problem we turn to pertinent literature.

Data pertaining to the kinematics of point-to-point arm movements have consistently revealed relatively symmetrical and bell-shaped speed profiles of the hand[4] (Abend et al., 1982; Atkeson & Hollerbach, 1985; Flash & Hogan, 1985). In taking advantage of this descriptive property, it is hypothesized that such a profile can be imposed on each joint as it moves from its starting angle to its target angle. All joints, furthermore, start and end their movement together. By doing so, the resulting hand velocity profile is bell-shaped, and the cumulative amplitude covered by each joint increases sigmoidally from 0 at t=0 to α_j at $t=MT_j$.

It is important to note that the absence of a more complicated algorithm to compute postural transitions in this model merely reflects a belief that, compared to the inverse kinematics problem, movement generation is of secondary importance to the system. In other words, what the system "cares for" is postures. Once the appropriate target posture has been selected, and given that efficiency is of major concern in this theory, a straight-line motion through joint-space is simply followed. The resulting movements are, therefore, by definition always monotonic.

Learning

The third and final component of the Knowledge Model involves learning, as a direct outcome of the way postures are assumed to be stored and manipulated. As mentioned before, the actor is assume to possess a pre-specified number of stored postures. There are, therefore, two factors that may be directly manipulated, namely the number of elements and the contents of the knowledge-base.

Number of Stored Postures

While work on the Knowledge Model has typically not addressed developmental issues, the effect of changing the number of stored postures in an actor's knowledge-base has been a topic of investigation. The direct outcome of allowing more postures is on the amount and variability of input that enters the weighted summation, which in turn yields the target posture. In other words, more postures signify more information about alternative ways in which one can reach for the same spatial location. As a result, the more the input, the more the possibilities for reaching in ways that are more accurate and potentially more efficient.

This effect is reflected in Figure 5. In Panel A, the mean number of feedforward correction cycles required to achieve a specified spatial accuracy (acceptable distance A = 3 pixels; approximately 0.45 inches) is shown to vary as a function of number of stored postures. The number of correction cycles, as mentioned before, reflects planning time. Here, it is shown that planning time decreased with the increase in number of posture representations in the knowledge-base. In Panel B, the mean spatial error cost resulting after the first planning cycle (i.e., no feedforward correction took place) is shown as a function of number of stored postures. The spatial error cost reflects the spatial accuracy achieved. Here, an increase in the number of stored postures allowed for higher spatial accuracy.

Contents of Knowledge-Base

Given our primary interest in adult-like behavior, our simulations are carried out with a fixed number of stored postures (typically $n = 600$). As mentioned in the beginning of this chapter, however, the contents of the knowledge-base are not assumed to be fixed. By making this assumption, we can simulate the development of expertise. This is achieved by the process described next.

Each newly generated posture is assigned a strength factor, $S(P_i)$, initially set equal to some base-value, or threshold, ω. As posture planning takes place, and each stored posture acquires weights reflecting its suitability for various tasks, its strength may change:

$$S(P_i)_{(t)} = S(P_i)_{(t-1)} + (1-\lambda)\, g(P_i)_{(t)}. \tag{13}$$

In this equation, $g(P_i)_{(t)}$ denotes the weight of stored posture P_i after task t, and λ is a constant ($0 \leq \lambda \leq 1$). A stored posture's strength, therefore, increases depending on how helpful it has been for posture planning. It does not have to have been adopted per se for its strength to increase, but has to have made some valuable contribution to the weighted sum which subsequently yields the target posture.

After each task is completed, all postures' strengths are compared to the threshold, ω. Stored postures whose strength is smaller than ω are discarded and replaced by new, randomly chosen postures. The implications of such a process are depicted in Figure 6. Here, an actor who started out with a randomly distributed knowledge-base of stored postures (Panel A: note the non-linearity of the mapping between postures and end-effector locations) was repeatedly presented with targets at arm's length away from the body and at shoulder height. After 600 presentations of such target locations, the end-effector locations were plotted again (Panel B). It is now evident that a "migration" of stored postures took place, allowing for the development of expertise in the area of repeated learning. This effect is a direct outcome of the changes in the knowledge-base that take place in the manner described above.

Evaluation

The Knowledge Model has been evaluated and found to qualitatively account reasonably well for a large body of data concerning simple reaching tasks in unobstructed environments. It has been shown to predict and explain numerous phenomena: compensation for changes in the mobility of joints, timing effects, reaching with hand-held tools, the development of expertise, and sequencing effects. In fitting the model's performance to data collected from human subjects the model has been found to account for more than 98% of the variance in observed target postures (Vaughan, Rosenbaum, Loukopoulos, & Engelbrecht, 1993a). In addition, the procedure of fitting the model to experimental data has allowed for a preliminary determination of the model's parameter values. All the simulations I will be referring to from here on, therefore, have been produced with these parameter values which are listed in Table 1. For more details on the model's performance, however, the reader is again referred to the theory's exposition (Rosenbaum et al., 1993a, b; Rosenbaum et al., 1993c).

CHAPTER 3

OBSTACLE AVOIDANCE

Overview

In working closely with the Knowledge Model, I have become increasingly concerned with the simplifying assumption that the actor's work-space is always devoid of obstacles. An important characteristic of our own environment, however, is that it is highly complex, non-static, and cluttered. In fact, even in the absence of physical obstacles, our own bodies present objects around which we have to move in order to avoid collision. And yet, we all possess the remarkable ability to carry out complex movements in such demanding environments. We appear to do so, most of the time anyway, relatively effortlessly and successfully. This theory, therefore, is incomplete if it does not offer an explanation for this extraordinary ability. In addition, I believe that all future extensions of the theory (e.g., those that account for grasping objects) will inevitably require some explanation of the way people move around obstacles. My thesis, therefore, is meant to address the question of how human beings reach in the presence of obstacles.

Literature Review

Obstacle avoidance problems became particularly manifest with the development of computer-controlled manipulators during the late sixties (Udupa, 1977). It is not surprising, therefore, that most of the approaches to obstacle avoidance come from the field of computer science and robotics. While literally thousands of researchers have been preoccupied with finding solutions to this problem, I present the work of but a few. This review, therefore, is by no means exhaustive. Rather it attempts to give the reader a relevant background without getting too deeply involved with foreign terminology and computational complexities. As will become evident later, these papers contain ideas which have directly inspired and influenced my work and so tie well with the scope of this thesis.

Computer Science & Robotics

Udupa (1977)

Udupa's approach concerned the planning of safe trajectories for computer-controlled manipulators moving in 2 or 3 dimensions with two movable links and multiple degrees of freedom. The position and orientation of the end-effector in the goal configuration, as well as the locations and shapes of all obstacles were input by the user. In the first stage of the proposed method, Udupa used a series of decomposition methods to reduce the complexity of the real world representation of the manipulator and its work-space. First, the manipulator was represented by a single line segment in a *primary problem space.* Next, the manipulator was represented by a single point in a *secondary problem space.* The construction of these problem spaces of increasing abstraction resulted in an overall simplified representation of the manipulator.

In the second stage, Udupa took a hierarchical approach to planning. Here, after the goal configuration had been evaluated to be safe, the intermediate parts of the trajectory were planned first (*mid-section planning),* followed by planning of the terminal phases of the trajectory (*terminal phase planning).* At each stage, planning heuristics were applied hierarchically, first for the more proximal, then for the more distal links. Trajectory planning, finally, was primarily done in primary problem space while the secondary problem space was used, when necessary, for simplifications.

Mid-section planning was carried out by employing a curve approximation algorithm, first for the proximal, then the distal link trajectory. This algorithm approximated a curved motion by a series of small, connected straight-line motions. Every time a motion (from A to B) was judged to induce collision, a subgoal was introduced such that the single motion was segmented into two components (from A to P, and from P to B). The algorithm was applied recursively, until most of the path between starting and goal configurations was deemed safe. Terminal phase planning, finally, used a sequence of adjust and move motions to first position the proximal link in a safe location, and then try to manipulate the position of the distal link such that the chances of collision during the subsequent move motion were reduced.

Udupa's work was especially important because it presented one of the first attempts to tackle the collision avoidance problem. Problems which became apparent with this approach were addressed by other authors in later years. Some such problems are that of requiring the goal configuration input, the over-reliance on simple heuristics which frequently lead to impasses, and the non-optimality of the yielded trajectories.

Lozano-Perez (1983)

The focus of Lozano-Perez's work was an algorithm for computing constraints on the position of a manipulator due to the presence of other objects. Since its appearance in 1983, it has been the major basis of the work of many computer scientists and roboticists attempting to solve the collision avoidance problem. This work introduced the *configuration-space*, a space whose coordinates represent the degrees of freedom of the manipulator. Within this space the position and orientation of an object can be represented by a single point.

Two major types of problem were addressed. The first, the *Findspace* problem, lies in determining where an object A can be placed inside some region so it does not collide with any other object B in the same region. The second, the *Findpath* problem, consists of finding how to move the same object A from one location to another in the specified region without causing collisions with other objects B_j. Lozano-Perez showed that both of these geometric problems could be solved simultaneously in two steps: by building a data structure that captures the geometric constraints of the work-space, and by searching the same data structure to find the appropriate solution.

Initially, a data structure — representing the configuration-space — was created by allowing each coordinate of that space to represent a degree of freedom in the position or orientation of object A. Within this data structure, it was possible to build geometric objects, called *configuration-space obstacles*, that represent all the positions of A that caused collisions with the obstacles B_j. Given this formulation, the two problems stated above were simplified to finding a single point (representing a "safe" position of A), or path (representing a series of "safe" positions of A linking its initial and goal positions) outside the configuration-space obstacles.

The configuration-space approach was shown to solve a variety of Findspace problems. Less emphasis was placed on the Findpath problem, which was described as a graph search problem and, hence, relied on heuristics. Its solution, therefore, was not optimal in many aspects (e.g., it was sensitive to inaccuracies of the configuration-space and yielded paths that just touched the obstacles), and the approach failed to find optimal paths among three-dimensional obstacles. Although the relevant computations were complex and some solutions sub-optimal, Lozano-Perez's general approach has proven extremely useful; it will become the subject of discussion later in this thesis.

<u>Muthuswamy and Manoochehri (1992)</u>

Both Udupa's *hierarchical planning* approach and Lozano-Perez's *configuration-space* approach relied on heuristic searches of free paths. Such heuristics do not result in solutions that are optimal with respect to robot performance criteria (i.e., total travel time, smoothness of joint motions, or minimization of power consumed by the manipulator). This appeared to be a major concern for Muthuswamy and Manoochehri, who attempted to derive an approach integrating both simple heuristics and optimization techniques in deriving optimal path planning. Their methodology employed three steps: discretization of the cluttered work-space, construction of a network graph, and computation of the optimal path.

In the first step, a planar two link manipulator work-space was represented by a rectangular grid of discrete points. Extreme reaches of the manipulator lay along each coordinate of a grid whose resolution of all intermediate points was defined by the user. Within this space, the regions occupied by the obstacles were marked as inaccessible. Given the starting and goal configurations of the manipulator, an elliptical search-space was defined. This subset of the work-space included the starting and ending points — representing the starting and ending configurations of the manipulator — which lay at the foci of the ellipse. The size of the search-space was a parameter. Defining a search-space dramatically reduced the number of nodes to be examined in seeking an optimal path.

In the second step, the search-space was transformed into a network graph. First, connectivity between neighboring grid points was defined, so that all possible paths through the search-space were represented. To limit the scope of search for the optimal path, however, an angular deviation constraint between the line connecting the current point and a potential neighbor, and the line connecting the current point and the end point was set. Consequently, only neighbors satisfying this constraint were said to be connected. Thus, given a high angular deviation constraint, the number of possible paths was substantially reduced.

In the third and final step, two objective functions were defined: minimization of total travel time and minimization of a measure of mechanical work. Each connection between neighboring grid points, therefore, could now be weighted with respect to these two objectives. Selecting the optimal path, according to the authors, could make use of any optimization technique; in this case they employed a minimum cost algorithm. The latter, known as Dijkstra's minimum cost algorithm (Minieka, 1978) finds the minimum cost path between any two points in a search-space by iteratively setting each grid point's weight equal to the minimum of its neighbor's weights. When no weight changes at a given iteration, the process stops and the optimal path is defined by following the points which have acquired minimum weights.

The success of this algorithm was alluded to, based on its commercial application in planning optimal paths for SCARA (Selective Compliance Assembly Robot Arms) robots. Note, however, that as in the previous approaches, this algorithm required the input of both start and goal configuration without addressing the question of how these are computed. Also, a problem common to space discretization lies in the resolution of the graph. The trade-off between successful collision avoidance and resolution of the network graph, which is user-specified, is therefore problematic. Setting a low resolution to avoid excess demands on the representation of the work-space, is more likely to lead to an inaccurate description of the space, which in turn increases the chances of collision.

Connolly and Grupen (1993)

From the Computer Science Department of the University of Massachusetts came a similar approach to the Findpath problem. This approach, once again, made use of the notion of configuration-space. The details of mapping the work-space onto a network of nodes with neighbor-like properties was similar to that described by Muthuswamy and Manoochehri. In the case of Connolly and Burns, however, a different optimization technique was used to find a collision-avoidance path.

The technique presented by these scientists required that a harmonic function — a solution to Laplace's Equation — be computed over the free regions of configuration-space[5]. By assigning high potentials to obstacles, low potentials to goals and repeatedly averaging the free points in the interior so that their potential equals the average of their neighbors at each iteration, the values of the free areas eventually converge to a harmonic function, thus giving rise to a collision-free trajectory of the manipulator. Based on relevant research regarding the underlying properties of various brain regions, these computations were assumed to take place in the striatum and specifically in the basal ganglia (Connolly & Burns, 1992, 1993)

This work was particularly interesting because of its attempt to render it neurologically plausible. While elegant and simple, however, paths computed in this way were not always optimal with respect to possible energy-constraints. In addition, no indication was given of how connectivity between neighboring points may be determined. Finally, once again, no reference was made to the selection of starting and goal states.

Kawato, Maeda, Uno, and Suzuki (1990)

Kawato and his colleagues explained trajectory planning and control of arm movements between a starting and a goal state in terms of dynamic optimization. In their approach, an objective function was expressed as a time integral of a performance index. The system was assumed to optimize performance with respect to this index which, in this case, is the square of the rate of change of torque integrated over the total time for the movement[6] (Uno et al., 1989). Two kinds of constraints were imposed on the movement: hard constraints such as the starting, ending, and the via (if required) configurations, and soft constraints such as the smoothness of movements. To achieve obstacle avoidance, the authors included the necessity of a body configuration not to induce collision as a hard constraint in their system.

In its neural network implementation, the *minimum torque-change* model employed a relaxation method to derive optimal via point trajectories by repeatedly evaluating and changing the postures comprising the trajectory at each time step. One problem, therefore, lay in the determination of the number of time steps (in other words, the resolution), which was a user-specified parameter. Also, the evaluation of this model rested upon two-*DOF*-freedom simulations (a non-redundant problem) so it is not clear that it would be easily extendible to more degrees of freedom. Finally, no details were given about the computation of the hard constraints imposed by the obstacle, or the determination of the appropriate via configuration.

<u>Summary</u>

In summary, two routes have traditionally been suggested as solutions to the obstacle avoidance problem. The first (the configuration-space approach), takes advantage of alternative ways to represent the problem space such that the appropriate path can be prescribed more simply. This approach has the disadvantage of not addressing the issue of optimization of paths in terms of dynamic constraints. The other approach (the network approach), again attempts to represent the problem space in simpler terms, while also taking into consideration dynamic constraints. However, it has the disadvantage of requiring networks of large resolution and is subject to local minima. Let us now turn to the approaches taken by researchers in the field of psychology.

Diamond (1990)

In a series of experiments with infants ranging from 5 to 12 months old, Diamond investigated the onset of what can be described as obstacle avoidance. The task she employed consisted of placing a toy inside a transparent box which is open on one side. Infants until the age of about 7.5 months failed to retrieve the toy from within the box. Diamond suggested that the failure of retrieval was largely due to the inability to inhibit certain motor reflexes, such as that of grasping (grasp reaction) or withdrawing the hand (avoidance reaction) as soon as the first surface was touched, which in these cases was the box surface. Later on, when infants were more capable of inhibiting such reflexes, the retrieval failure was still evident. This time, Diamond concluded that the problem was due to the failure to inhibit reaching straight for an object. It is not until 8.5 - 9 months of age that infants were able to separate their line of sight from the line of reach and so, by moving their head and body to acquire a better view of the situation, were able to conceive of and perform the necessary detour to retrieve the toy.

These studies, along with their follow-ups (Diamond, 1993), provided an insight concerning the onset of obstacle avoidance. They demonstrate that such an ability is based on complex cognitive processes which, in turn, depend on the maturation of specific brain areas (the supplementary motor area (SMA) of the frontal cortex). Specifically, I believe that they suggest that obstacle avoidance requires the interplay of self-awareness, advanced visual perception, and a high degree of skill.

Warren (1984)

A critical question related to the study of obstacle avoidance is how perception and action are coupled. From an ecological perspective, as first proposed by J. J. Gibson (1958), comes the idea that behavior is visually guided by perceiving what the environment offers or affords for action. An affordance, therefore, is the functional utility of an object for an animal with certain action capabilities (Warren, 1984, p. 683).

Many experiments designed to investigate adults' perception of affordances have been carried out, and have demonstrated that people can detect the maximum height of obstacles onto which they can step (Warren, 1984), sit (Mark, Baillett, Craver, Douglas, & Fox, 1990), as well as the width of apertures through which they can locomote (Warren & Whang, 1987). This ability is apparent in both adults and children. In addition to limits of action, however, people are also able to detect the most efficient paths of action.

A representative series of experiments investigated affordances in human stair-climbing (Warren, 1984). In a first experiment, Warren showed that both short and tall subjects judge stairways as unclimbable at a riser (step) height equal to some constant proportion of their leg length. In a second experiment, visually preferred riser height was found to be predicted by the directly measured energetically optimal riser. These findings were also successfully replicated in children (Pufall & Dunbar, 1992).

Such findings implied that there is an intrinsic or "body-scaled" metric on which people base their perception of the environment and its affordances. They suggested that, by accurately perceiving critical and optimal points, people are able to assess their work-space and adaptively alter their actions when and if necessary. Perception and action are, therefore, closely coupled; our perception of the world is scaled in terms of the biomechanical and physical limits of our own systems of action. These ideas, as we will see in the next chapter, will become very important in the development of an obstacle avoidance algorithm.

Engelbrecht and Rosenbaum (1993)

Engelbrecht and Rosenbaum recently proposed a model of movement planning in the presence of obstacles. While bearing resemblance to Muthuswamy and Manoochehri's (1992) work outlined above, this approach was developed independently and was formulated in terms of a neural network. The implicit constraints were also different: Engelbrecht and Rosenbaum relied on minimizing energy expenditure and achieving collision avoidance. Their network also employed an adaptation of Dijkstra's algorithm for finding the shortest path in a digraph.

Engelbrecht and Rosenbaum's approach was evaluated by means of a computer simulation, and was shown to be generally successful. Theoretically, it was able to find optimal paths regardless of the number of obstacles or the dimensions of the work-space. However, it was a purely kinematic model which did not address the important issue of how motion could be generated given a specified path, and was generally confined to the elements represented by the network. In other words, the issue of resolution, once again, can plague the system, while the movements are restricted to those represented by the network (i.e., no novel solutions —postures— can be adopted).

Dean and Brüwer (1992)

In one of the few experiments directly examining obstacle avoidance, Dean and Brüwer reported on the observed characteristics of human arm movements in the presence of a linear obstacle. All movements were conducted in the horizontal plane and employed the shoulder, elbow, and wrist. Subjects were required to move a pointer between two spatial locations while avoiding an obstacle of varying length lying on the horizontal surface of movement. In the absence of an obstacle, hand paths were found to be straight with underlying bell-shaped velocity profiles, as reported elsewhere in the literature. When an obstacle was present, however, the hand paths became curved, with velocity profiles that frequently had two peaks. The valley between peaks corresponded to the part of the path where the curvature was greatest.

The authors concluded that the paths observed in the obstacle avoidance conditions were similar to those used when making movements involving via points. The decrease of hand tangential velocity at points of maximum curvature, furthermore, was considered to be evidence for the segmentation of movements. In other words, the movements seemed to be composed of a series of sequential straight-line movements. The fact that the minimum distance of the hand from the obstacle, finally, was constant for obstacles of different lengths, suggested that minimum distance may be a planning constraint.

Conclusion

After reviewing the work described above, it becomes clear that obstacle avoidance is a very hard problem. While numerous approaches have been taken over the years, none has proven to combine simplicity with success. In addition, none has simultaneously addressed the Findspace and the Findpath problems in ways that can be directly applied in the context of human reaching control. This fact makes the problem interesting and challenging. Specifically, it suggests that some of the outlined procedures can be incorporated into the Knowledge Model, thus allowing it to account for obstacle avoidance.

CHAPTER 4

OBSTACLE AVOIDANCE AND THE KNOWLEDGE MODEL

The Degrees of Freedom Problem Revisited

In the beginning of this thesis I introduced what I presented to be one of the major problems faced by students of motor control: the degrees of freedom problem. The fact that joint redundancy characterizes the human body, as one might expect, is not accidental. To understand this, a simple example is presented in Figure 7. Here, the same stickfigure as in Figure 1 is shown reaching with the hand for the same spatial location in the absence (Panel A) and in the presence of an obstacle (Panel B). Notice that, although in the first case the choice of body postures is infinite, this redundancy is reduced in the second case because of the obstacle. In this latter case, the obstacle presents an additional constraint on the choice of appropriate target postures. As opposed to just a problem, therefore, the choice between degrees of freedom can also be viewed as a desirable attribute for a system with enough flexibility and adaptability to operate successfully even in those cases where external constraints are present.

Outline

The Knowledge Model presents a tool upon which to build a new theory of obstacle avoidance because it was designed based on the degrees of freedom problem. One way to attack the obstacle avoidance problem, therefore, is to re-examine the Posture Planning component of the model. In doing so, I will attempt to show that selection of a target posture in view of the presence of an obstacle can be the first step in successful obstacle avoidance. The necessary modifications will take into account the possible constraints posed by the obstacle, along with the problem of deciding which way to go around the obstacle. Selection of an appropriate target posture, however, will not necessarily guarantee a collision-free postural transition. The next step in attacking the obstacle avoidance problem, therefore, will be to modify the Movement Execution model component. Here, the need for non-monotonic paths will be demonstrated, along with ways in which these can be produced. To achieve the latter, I will postulate the need for via points and will propose ways in which the movements through via points can be produced and manipulated.

Model Components

Posture Planning in the Context of Obstacle Avoidance

Posture Planning can be thought of as a solution to the Findspace problem. We may, thus, turn to the notion of configuration-space. Recall that the basic idea is to convert the problem situation into a space in which the modeled system may be thought of as a point. In the case of the Knowledge Model the relevant space is posture-space. Figure 8 (Panel A) represents a stickfigure representation of an arm with joints J_1 (range of motion 180-270°) and J_2 (range of motion 0-180°). The configuration-space, therefore, is two-dimensional, each dimension — or axis — representing the working range of a given joint (Panel B). Each point in this space represents a particular configuration of the arm, so, for example, the configuration in Panel A is marked by a dot on the configuration-space in Panel B. The constraint imposed by the presence of an obstacle is that it "blocks off" certain parts of the configuration-space. In other words, certain combinations of joint angles bring some part of the arm into collision with the obstacle. These combinations are represented by shaded areas in configuration-space. By extension, it is possible to assess the posture-space in view of a particular situation, by finding those stored postures which, if adopted, would induce collision.

Assessing the Possibility of Collision

Computation and assessment of the appropriate configuration-space, as presented in Lozano-Perez's paper (1983), can be a complex mathematical procedure. In the case of the Knowledge Model, however, all that is required is an additional computation at the stage where stored postures are evaluated for their suitability to a reaching task in question. A linear interpolation method is employed to determine the possibility of collision.

Consider two line segments with origins at (O_{1x}, O_{1y}) and (O_{2x}, O_{2y}) respectively, and lengths of (D_{1x}, D_{1y}) and (D_{2x}, D_{2y}). To determine if the two segments intersect, it suffices to find a single point common to both. The following equations define the Cartesian locations of any possible point p_1 belonging to the first line segment:

$$p_{1x} = O_{1x} + \alpha D_{1x} \qquad \text{and} \qquad (14)$$

$$p_{1y} = O_{1y} + \alpha D_{1y}. \qquad (0 \leq \alpha \leq 1) \qquad (15)$$

Similarly, the following equations define any possible point p_2 belonging to the second line segment:

$$p_{2x} = O_{2x} + \beta D_{2x} \qquad \text{and} \qquad (16)$$

$$p_{2y} = O_{2y} + \beta D_{2y}. \qquad (0 \leq \beta \leq 1) \qquad (17)$$

Solving for the two factors, α and β, makes it possible to determine whether there exists a common point belonging to both line segements. By equating (14) and (16) we solve for α:

$$\alpha = \frac{(O_{2y} + \beta D_{2x} O_{1x})}{D_{1x}}, \qquad (18)$$

and by equating (15) and (17) and substituting (18) for α:

$$\beta = \frac{((O_{2y} D_{1x}) - (O_{1y} D_{1x}) - (O_{2x} D_{1y}) + (O_{1x} D_{1y}))}{((D_{1y} D_{2x}) - (D_{1x} D_{2y}))}. \qquad (19)$$

If both factors are between 0 and 1.0, then the two line segments intersect; otherwise, they do not. In a similar fashion, to determine if a stored posture brings any limb into contact with the obstacle, we simply compute the two factors for each of the limbs in the orientation specified by the posture under evaluation, and for each of the sides of a rectangular obstacle so that all the limbs are checked against all of the obstacle sides[7].

Once the possibility of collision has been determined, those stored postures that are evaluated to be collision-inducing are allotted maximal total costs (equal to 1.0) and are excluded from the weighted sum. Recall that a total cost of 1.0 is placed at the rightmost end of the abscissa where the Gaussian costs associated with it are essentially equal to zero. The weights assigned to these postures are minimal, and their contribution to the determination of the target posture is, therefore, also minimal. On the other hand, postures which are found to be collision-free are subjected to the usual evaluation based on spatial error and travel cost, and are subsequently entered into the computations for a target posture.

Partitioning the Work-Space

While the method outlined above allows for the derivation of collision-free target postures, it does not guarantee it. The reason is that the resulting target posture is an average of stored postures. An interesting example is the case where averaging points around a known shaded area in configuration-space yields a point exactly between them, and thus within the shaded area. In those cases where the target posture is not suitable, an additional procedure is assumed, one which requires *partitioning* of the work-space.

The idea of partitioning the work-space is based on the notion that it is natural to categorize which side of an obstacle to move around, when the obstacle is in the way of reaching for a spatial location. It is assumed, therefore, that people are somehow able to construct a cognitive representation of the problem situation in terms of its geometrical properties and what it affords. This representation can include constructs such as "above," " below," "to the right of," and "to the left of" an obstacle, similar to the constructs of a critical limit and efficient path of action, which we have seen people can compute (Warren, 1984). In addition, studies conducted in our laboratory have revealed a certain consistency with respect to the choice of movement direction around the obstacle, depending on the movement condition. More will be said about the experimental design in question in the section entitled Qualitative Fit to Data in the last chapter.

The heuristic to perform the partitioning, presented below, is merely a computational convenience. No claim can be made, at this point, about the way it is actually carried out in the brain. If, however, people are indeed able to somehow derive these constructs, then we may attempt to do the same simply by partitioning the contents of the knowledge-base into similar groups.

Reaching around an obstacle in two-dimensional space can be achieved in two ways: reaching "above" or "below" the obstacle. We thus attempt to separate the knowledge-base into two such groups. To determine which side of the obstacle the body would result in, given a posture P_i, it is first necessary to define an axis of partitioning, the JTO line (Joint To Obstacle line). This axis is an imaginary line extending from the first free-moving joint (the hip) through the center of the obstacle[8]. Each stored posture has to be evaluated with respect to this line or some part of it.

Posture P_i is classified as "above or "below" the obstacle as follows. First, the extrinsic position of the next joint (the shoulder) is compared with respect to the JTO line. This results in the first classification of P_i as "above" or "below." Next, the linear interpolation method (Equations 14-19) is employed to determine whether the upper arm segment intersects the line *segment* connecting the hip and the center of the obstacle. If the two line segments intersect, the classification is reversed. If they do not, posture P_i's characterization remains the same. This last process is repeated until all limb segments have been evaluated, and the final classification is the one that determines the grouping of posture P_i.

Figure 9 shows an example of the outcome of this process. Here, posture A was judged to be an "above" posture because the shoulder is above the JTO line, and none of the limb segments intersect the JTO line segment. Posture B was judged to be "below" because, while the shoulder lies above the JTO line, the upper arm intersects the JTO line segment.

In those cases where partitioning of the work-space is required, the weighted summation process can be carried out twice, once for each group. This, in turn, yields two different target postures. The two then compete to yield the single, most appropriate target posture for the task in question. Once again, it is important to fulfill the demands of the task, so the two target postures may compete on the basis of spatial accuracy (spatial error cost), if this is most important, or on the basis of efficiency (travel cost), or even on the basis of both costs (total cost).

The heuristic for classifying postures can also be applied to the resulting target posture. This explains how the actor may know which side around the obstacle to move, before planning the actual movement. I term this side the <u>avoidance side</u>. Finding the avoidance side simply requires examining the target posture with respect to where it brings the body —above or below the obstacle. The importance of this information will become evident in the next section.

I have now made all the necessary modifications to the Posture Planning component of the Knowledge Model. I have taken advantage of all the information provided to us "for free," and so have imposed little extra computational demands on the system. Selecting the appropriate goal configuration, however, does not guarantee a safe postural transition. For this, let us turn to the next model component.

Movement Execution in the Context of Obstacle Avoidance

A major premise of the Knowledge Model is inherent in the type of joint trajectories generated by the Movement Execution component. As outlined above, the theory predicts that each joint travels from its starting angle to its target angle through the shortest possible path in joint-space. All joints, furthemore, are assumed to start and end their movement together. Their respective trajectories, therefore, are by definition monotonic. In other words, no joint reverses its direction of movement during its motion.

One of the intuitions concerning reaching around obstacles is that movements are less straight than they would otherwise be. In effect, formal (Flash & Hogan, 1985; Dean & Brüwer, 1992) and informal observation suggests that non-monotonic paths of the hand in extrinsic space are almost always involved when obstacles are avoided. This is shown in Figure 10 where the stickfigure is shown to have selected the appropriate target posture based on the algorithm described above, but fails to generate a collision-free path because the chosen movement was computed to proceed in a straight line through joint-space without taking into account the possibility of collision. Appropriate modifications of the Movement Execution component are, therefore, required.

Non-monotonic trajectories in the context of the Knowledge Model can be simply and naturally induced by positing movements through via points. In other words, such trajectories can be generated if the actor is required to move, without stopping, through a spatial location prior to reaching the desired destination. The idea of via points for obstacle avoidance has been suggested before (Flash & Hogan, 1985; Uno et al., 1989). A review of the literature, however, has failed to suggest ways in which the spatial location of the via point is determined when obstacles must be avoided.

Finding the Via Point Location

What is the appropriate spatial location of the via point such that it induces a collision-free trajectory? The geometrical properties of the work-space can, again, provide useful information. I believe that the important factor in this problem lies in finding the one, most protruding point of the obstacle given the starting and target posture of the actor. If the actor succeeds in avoiding the most protruding part of the obstacle, s/he can avoid collision with the obstacle.

The appropriate geometrical computations to find the most protruding point of an obstacle require that two beams be drawn from both the starting posture hand location and the target posture hand location. The beams are rotated until they come into contact with the obstacle on the avoidance side. Recall that the latter was determined during Posture Planning. In the case of a circular obstacle, these beams lie tangent to the perimeter of the obstacle (Figure 11, Panel A). In the case of a rectangular obstacle, they touch the corners of the rectangle (Figure 11, Panel B). A line bisecting the angle between the two beams defines a third line called the Via Point line. It is postulated that the via point lies on the Via Point line, some distance away from the intersection of the two beams.

This distance, the Clearance parameter, is dependent upon factors such as the material properties of the obstacle and the task demands (e.g., speed requirements).

Finding a Via Posture

Motion through a via point before reaching a final target destination can be thought of as two separate movements, one which first brings the hand from its starting location to the via point, and another which brings the hand from the via point to the target. To compute the two movement components, therefore, it is first necessary to find a Via Posture, $\bar{P}_v$. This is achieved in a manner similar to that of determining the target posture.

Each stored posture is first evaluated with respect to whether it brings the body into collision with the obstacle. Those postures that do induce collision are assigned a total cost of 1.0 (maximum value). The remaining postures are considered for their spatial error and travel costs. Their weighted sum yields the via posture. The movements from start to via and from via to target posture, however, cannot be treated as independent because of the sequential nature of the task. That is, the model must account for the fact that the way in which a person will reach for the first target (the via point) will differ from the way they would reach for it if no subsequent movement (to the target location) was required. In Figure 12, the stickfigure is shown reaching for the two spatial locations, in a sequential (Panel A) and in a non-sequential fashion (Panel B). Note how the posture adopted at the first target (the via point, V, in Panel A) is markedly different from that adopted at the first target (T_1 in Panel B). Such anticipation effects have been demonstrated in the context of various motor control acts, such as reaching (Rosenbaum, et al., 1992; Fischer et al., 1993), speaking (Moll & Daniloff, 1971), and typewriting (Gentner, Grudin, & Conway, 1980).

To account for anticipation effects, the computation of the travel cost for a posture, P_i, is slightly modified. Postures being assessed for bringing the hand to the via location must be evaluated with respect to the energy they would require to do so when departing from the starting posture, as well as when subsequently moving to the already-determined target posture. The travel cost for posture P_i, therefore, is the average of two travel costs: that of moving from the starting posture, and that of moving to the target posture. The calculations for the component travel costs are those described in Equations 4-7.

The total cost of each non-collision-inducing stored posture, finally, is computed by taking the weighted sum of the spatial error and the combined travel costs:

$$C(P_i) = w_d \left(\frac{D(P_i)}{MaxD} + w_v \left(\frac{\frac{V_1(P_i)+V_2(P_i)}{2}}{MaxV}\right), \qquad (20)$$

where $V_1(P_i)$ is the travel cost for moving from the starting posture to posture P_i, $V_2(P_i)$ is the travel cost for moving from posture P_i to the target posture, and $D(P_i)$ is the spatial error cost. Based on their total costs, postures are assigned weights as described above, in the section labeled Weights. By taking the weighted sum of all stored postures, as in Equation 11, it is now possible to compute the Via Posture.

Moving Through a Via Point

Another point to be addressed in this obstacle-avoidance algorithm pertains to the movement through a via point en route to a target destination. A way to achieve such a transition is inspired by research employing a double-step target-switching paradigm (Flash & Henis, 1991; Henis & Flash, 1992). Here, subjects were instructed to start moving to a target location as soon as it appeared. At varying times after movement onset, however, and without prior warning, the target location changed. This line of work, therefore, investigated the instantaneous modification of underlying motor plans in response to a sudden switch of the target location. The researchers modeled these movements based on the minimum-jerk principle, and postulated that when the target switches spatial locations the initial motor plan is not aborted. Instead, it is allowed to carry on until its completion. In addition, however, a second motor plan is created in response to the target switch and is vectorially added to the first plan. The resulting movement retains its smoothness and is still characterized by a bell-shaped, albeit sometimes bimodal, velocity profile. The results of their modeling supported these ideas.

The target-switching paradigm is admittedly different from the situation of having to move through virtual via points in order to avoid collision. In the latter case, the actor has complete knowledge of the two target locations before developing one or more motor plans. The principle, however, lends itself nicely to the purpose of moving through via locations without actually stopping. It can be posited that the actor vectorially sums two motor plans (trajectories), from start to via, and from via to target, thus achieving a natural transition without stopping at the via. A related question, therefore, is that of determining the time at which the second component starts being vectorially added to the first.

Vectorial Summation of Two Motor Plans

The time at which two component movements begin to be overlaid has major implications for the qualitative characteristics of the resulting movement. In Figure 13, two hand paths (start to via and via to target) are shown to overlap at times ranging from 0 to 100% of completion of the first. This percentage is termed the delay to reflect the delay after which the second component begins to be added. First, the path prescribed by the hand is shown to be markedly different depending on the delay (Panel A). In the case of a 10% delay, for example, the hand path is almost straight but fails to go through the via location. In another case, where the delay is 60%, the hand path is curved and reaches the via location before moving on to the target location. Curvature, therefore, increases with the degree of delay.

Summing the velocity profiles (assumed to be bell-shaped) of the same two hand movements with varying degrees of delay also has major implications for the characteristics of the resulting velocity profile (Figure 13, Panel B). To take the same examples again, at a delay of 10% the velocity profile is bell-shaped and unimodal. At a delay of 60%, however, the profile is distinctly bimodal. As one increases the amount of delay, therefore, the velocity profile which is initially unimodal becomes bimodal. In conclusion, the degree of delay is an important control parameter, as will be illustrated later.

Checking Trajectories for Collision

Both via and target postures have, by now, been selected to be collision-free. The motor plan to achieve the necessary postural transition is determined as shown in the section labeled Movement Execution. Here, two movements are initially computed, one from starting to via posture, and another from via to target posture. Next, the two movements are overlaid. To conform with the principles underlying movement generation in the Knowledge Model, a very small overlap of the two movement components is initially imposed (e.g., a delay of 10%). This ensures that the resulting movement is mildly non-monotonic and the hand speed profile is unimodal.

Before carrying the movement out, however, it is necessary to assess its ability to avoid bringing any part of the body into contact with the obstacle. The movement execution algorithm of the Knowledge Model does not dictate that a movement is carried out overtly, but can be computed and stored until its execution. By viewing the transition between two postures as a series of independent postures, it is possible to assess the safety of any trajectory by determining if it is safe to adopt any and all of its components. For this, we apply Equations 14-19 to the posture corresponding to each of the discret time-steps comprising the trajectory.

We now turn to the final stage of the Obstacle Avoidance algorithm. What strategies can be adopted in the event that the trajectory is found to induce collision? Two methods are available. One method already mentioned is to vary the temporal overlap of the two movement components (i.e., vary the delay). Another is to vary the spatial location of the via point. These two methods are discussed below. If, after both these strategies have been tested, the resulting trajectory still induces collision, the task is deemed insoluble. Being able to come to such a conclusion prior to executing a movement is very important because informal observation, once again, suggests that humans are able to assess the feasibility of a task before attempting to carry it out.

Changing the Delay

The capability of subjects to produce movements with varying degrees of curvature has already been documented in the literature. In studies of horizontal point-to-point movements employing 2 degrees of freedom, subjects moved their hand in a straight line and with a speed whose profile was bell-shaped (Morasso, 1981). When, upon instruction, subjects were asked to produce curved hand paths, the movements produced were segmented, as if containing multiple straight-line components (Abend et al., 1982). Inter- as well as intra-subject variability was found among speed profiles which were bimodal, or even sometimes multimodal. These results are encouraging in that they suggest that humans are able to willingly produce movements of varying curvature whose underlying characteristics may be the result of temporally overlapping multiple movement components at different delays.

In a similar study, when subjects were instructed to move their hand from a starting location to a target location passing through a via point, the resulting hand paths were found to be curved (Uno et al., 1989). The velocity profiles, furthermore, depended on the location of the via point. If the via point was near the line connecting the start and end points, the velocity profiles were single peaked; if the via point was further away from the line connecting the start and end points, the profiles were double peaked.

A possible strategy to overcome an unsuccessful trajectory, therefore, is to vary the temporal overlap of the movements between start and via, and via and target locations. It is assumed that this is an exhaustive search of delays varying from 10% to 100%, and with a sufficiently large size of within-search increase (i.e., 10%). The implications of such a strategy have been outlined above, in the section labeled Vectorial Summation of Two Motor Plans. Varying the delay allows for trajectories of varying curvature. There exist cases, therefore, where changing the degree of delay provides a successful and computationally inexpensive alternative. An example of one such case is shown in Figure 14. Here, a delay of 20% of the two movement components (1 and 2) was unsuccessful in avoiding the obstacle as depicted by the hand path which goes through the obstacle. A delay of 30%, however, was successful.

Moving the Via Point

Changing the temporal overlap of two movement components may allow for movements of varying curvature; however, it can never result in a movement which allows the hand to move in an arc of greater curvature than the arc prescribed by the start, via, and target locations. The next best alternative to an unsuccessful trajectory, therefore, involves moving the via point. Recall that the via point is confined on the Via Point line, and its distance from the center of the obstacle is determined by a clearance parameter. As a last resort, therefore, and after varying the delay has proven unsuccessful, the clearance may be increased (i.e., the via point may be moved away from the obstacle and on the avoidance side). This method further enhances the exploration of the free space around the obstacle (Figure 15). Here, the cluster of potential trajectories whose hand paths are labeled A were created by assuming movement through V_1 and delays of 10-60%. These trajectories proved unsuccessful since they induced collision of some part of the body with the obstacle (as denoted by the solid parts of the hand path lines). The cluster of trajectories labeled B assumed movement through V_2 and also proved unsuccessful. Movement through V_3, however, and with a delay of 10% allowed the hand to reach for target T without collision of the body with the obstacle.

CHAPTER 5

CONCLUSION

Evaluation

Simulations

A first series of representative simulations is shown in Figures 16-19. Here, the stickfigure is shown avoiding obstacles in a variety of possible conditions. On the left side of each panel, no obstacle is present. On the right side, the same task is performed in the presence of an obstacle.

In Figure 16, cartoon representations of the stickfigure moving to avoid a small (Panel A) and a large obstacle (Panel B) are shown. Notice how the target postures adopted at the end of the obstacle-avoidance movements in both tasks were markedly different from those adopted for the corresponding tasks in the absence of an obstacle. This was a direct outcome of the modifications made in the Posture Planning component of the Knowledge Model. Note, however, that these are not the only observed differences. The trajectories in those cases where an obstacle was present were different from those in the corresponding cases where the obstacle was absent. On the left side of each panel, the movements followed a straight line through joint-space. On the right side, the presence of a via point induced non-monotonicity and the overall movement was the result of two sub-movements — from starting to via posture, and from via to target posture. The delay at which the two components were superimposed was small, so the resulting movement was mildly non-monotonic. These trajectories were the direct outcome of the additions made to the Movement Execution component of the model. Overall, however, the stickfigure moves with apparent naturalness.

Another interesting point to be made about these two tasks is that they required collision avoidance of different parts of the body. In Panel A, the lower arm was mostly at risk of collision; most changes in the presence of an obstacle, therefore, were made to the trajectory of the upper and lower arm while the torso remained relatively stable. In Panel B, however, the upper arm was also at risk of collision. To avoid the obstacle, therefore, the whole body was employed and the torso was also recruited to move the body backwards in order to safely bring the elbow around the obstacle.

In Figure 17, the stickfigure is shown to avoid rectangular (Panel A) and circular obstacles (Panel B). Once again, the trajectories were markedly different in the cases where an obstacle was present as compared to the corresponding cases where the obstacle was absent. The via point locations were determined as shown in Figure 11. Finally, the two tasks again required collision avoidance of differing parts of the body.

The next two Figures show the stickfigure avoiding an obstacle by moving under (Figure 18) or above it (Figure 19). Here, the importance of being able to compute which side of the obstacle the target posture brings the body around, is demonstrated: after computing a collision-free target posture, the avoidance side was determined such that appropriate positioning of the via point on that same side could be achieved.

Another important feature of the proposed obstacle avoidance algorithm is its ability to determine the insolubility of a given task. In Figure 20, one such case is shown. A target (T) was positioned near the end of the workspace. In the absence of an obstacle, a target posture (TP) was available for bringing the hand to this target. In the presence of an obstacle, however, this target posture brought the lower arm into contact with the obstacle. After employing all the heuristics described in Chapter 3 in order to find another, collision-free target posture, the task was deemed insoluble. That this is indeed the case can be demonstrated by plotting the two most extreme postures (A and B) that the stickfigure can adopt, given its angular rotation constraints. These two postures bring the body as close to the target as possible without allowing for contact with the obstacle. They both, however, fail to achieve the required spatial accuracy (acceptable distance A = 5 pixels, 0.75 inches) and are therefore not suitable. The target is unattainable.

Related Issues

Three-Dimensional Work-Space

As mentioned before, the Knowledge Model is meant to be applicable to systems requiring motion of more than just 3 degrees of freedom and in more than one rotational plane. As far as multiple *DOF* are concerned, postures can be simply modified to comprise of more than just 3 numerical entries representing the hip, shoulder, and elbow joint angles of the modeled system in this thesis. All computations are assumed to be performed in a parallel fashion, and so the toll on computational time should not be large, even if the modeled system contains 100 *DOF* as indicated by Turvey (1990a). Given that the world we move in is three-dimensional, however, it is imperative that a theory of movement selection and generation also address the issue of dimensionality. Postures have, until now, been represented by single numerical values representing the angular rotation of a joint in one rotational plane. The elbow angle entry, for example, consists of a numerical value representing the degree of flexion/extension of the lower arm in a sagittal plane. In a three-dimensional work-space, however, the elbow can flex/extend as well as pronate/supinate the lower arm. A single posture in three-dimensional posture-space, therefore, must contain two entries for the elbow: one for each of the dimensions in which it can rotate.

A related question, not yet addressed in the proposed obstacle avoidance algorithm, pertains to the partitioning of a three-dimensional work-space. It is envisioned, however, that much in the same way that people are assumed to possess the cognitive constructs of "above" and "below," they are also able to do the same in a three-dimensional context. In the latter case, and in addition to "above," and "below," the constructs could include "around the right side," and "around the left side" of an obstacle. By extension, four possible via point locations can be posited, and definition of the avoidance side based on the most suitable target posture allows for appropriate positioning of the via point.

Very Large Obstacles

As a final remark, I wish to address the possibility of obstacles extending further outside the work-space. Such a case, for example, frequently arises when one is seated at the table, with the hand in the lap, and wants to reach for an object on the table. In this situation, it is irrelevant whether the table extends out further than the 4 feet that are within reach. This is technically the case because, by definition, only "legal" postures are included in the knowledge-base. That is, given the seated position and the angular rotation constraints of the joints, only postures which bring the hand as far out as 4 feet are represented. It is postulated, therefore, that the obstacle avoidance algorithm also takes into account only the part of the table within reach. In defining the JTO line, therefore, the center of the obstacle is judged to be around the center of only the relevant part of the obstacle — the part within reach.

Future Extensions

While the major goal of this thesis, namely to suggest ways in which the Knowledge Model could be extended to explain obstacle avoidance, has been fulfilled, a number of issues still remain to be addressed. Below, are some of those I would like to pursue in the future.

Qualitative Fit to Data

A review of the literature has revealed an insufficiency of data pertaining to human obstacle avoidance. Recently, however, our laboratory collected such data. In this experiment, subjects were instructed to move between two spatial points in the presence or absence of an obstacle. The movements were carried out in the sagittal plane containing the subjects' right shoulder. Subjects were free to employ the hip, shoulder, and elbow joints, but were instructed to try to stay within the sagittal plane. Given the ease of access to this data, it is imperative that the performance of the new Knowledge Model be compared, at least qualitatively, to human performance. Two measures would be of interest: the avoidance side as it depends on task condition, and the qualitative fit of the stickfigure's hand paths to those observed during the experiment.

The first, the avoidance side for a given task condition, can be measured as the frequency of choosing to proceed above or below the obstacle when moving between start and target location. In fact, preliminary analysis of the data has already indicated a certain consistency with respect to the choice of avoidance side depending on target location. Simulations for the corresponding conditions, therefore, should show that the obstacle avoidance algorithm yields solutions that conform with the observed results. In other words, the stickfigure should choose to move in the same direction as the subjects were more likely to move, in the corresponding task condition.

Preliminary analysis of the data, once again, also indicated a large amount of within- and between-subject variability in terms of the observed hand paths. Most frequently observed were smooth hand paths with no evidence of movement segmentation but with underlying velocity profiles which were bimodal. In other cases, however, the observed movements were clearly segmented. The proposed obstacle avoidance algorithm, by virtue of its movement overlapping assumption, should be capable of simulating both conditions observed simply by varying the degree of delay. In an effort to provide a qualitative fit of the stickfigure's hand paths to those observed during the experiment, therefore, it is necessary to demonstrate that this is indeed possible. Naturally, the large variability observed in the data may be due to extrinsic factors, such as the instructions to the subjects, or the failure of subjects to conform to the instructions. At this point, therefore, more data would be necessary to further probe into human performance when avoiding obstacles.

Quantitative Fit to Data

Another way in which these data may be used is to show that the obstacle avoidance algorithm I have proposed yields solutions that are not only perceived to be natural and qualitatively similar to the data, but, upon further inspection, also replicate observed human behavior. The process of fitting the model to experimental data can ratify the model's strengths, as well as indicate possible limitations. More importantly, however, the question of regulating the location of the via point and the degree of delay for successful obstacle avoidance —as outlined at the end of Chapter 4— has been left quite open-ended. In the process of closely inspecting the data, I hope to become able to address such questions, and suggest a possible mechanism for appropriate control of these two factors.

Multiple Obstacles

Another important aspect in which the new Knowledge Model will have to be tested, is in its ability to successfully avoid multiple obstacles simultaneously present in the work-space. It has been envisioned that the same algorithm proposed in this thesis could apply in the presence of more than one obstacle. Theoretically, the constraints posed by the obstacles can be represented in the model's configuration-space in just the same way as when only one obstacle is present. In most cases, however, moving among many obstacles will require successfully avoiding one, then repeating the same process to successfully avoid the next obstacle, and so on until the final target destination is reached. These processes have not yet been fully worked out, and will obviously require more careful elaboration

Conclusion

I emphasized in Chapter 3 that the problem of explaining motion around obstacles is a difficult one. This is further complicated by the lack of relevant experimental data to guide and/or corroborate one's efforts. What I hope to have achieved, however, is to have built a computational model capable of generating questions and ideas about the kinds of experiments one would like to perform in the future. These experiments would be the best means for unveiling important information about an ability we all take for granted —obstacle avoidance.

Table 1: Parameter values as determined by fitting the model to experimental data.

Spatial Error Weight (W_d)	Hip Expense Factor (K_h)	Shoulder Expense Factor (K_s)	Elbow Expense Factor (K_e)
0.790 (maximum 1.0)	207.2 (maximum 222.0)	135.1 (maximum 222.0)	196.9 (maximum 222.0)

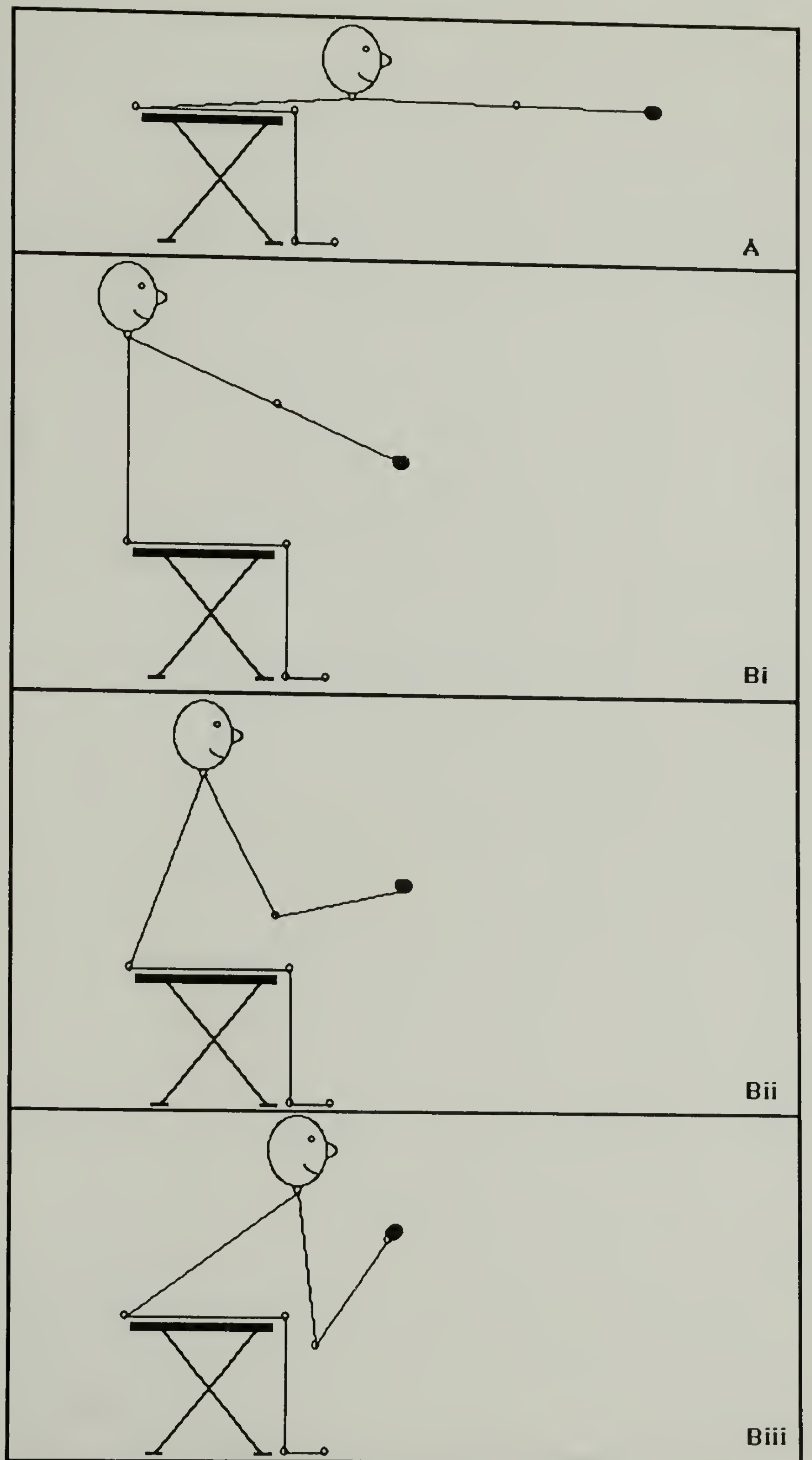

Figure 1: The degrees of freedom problem.

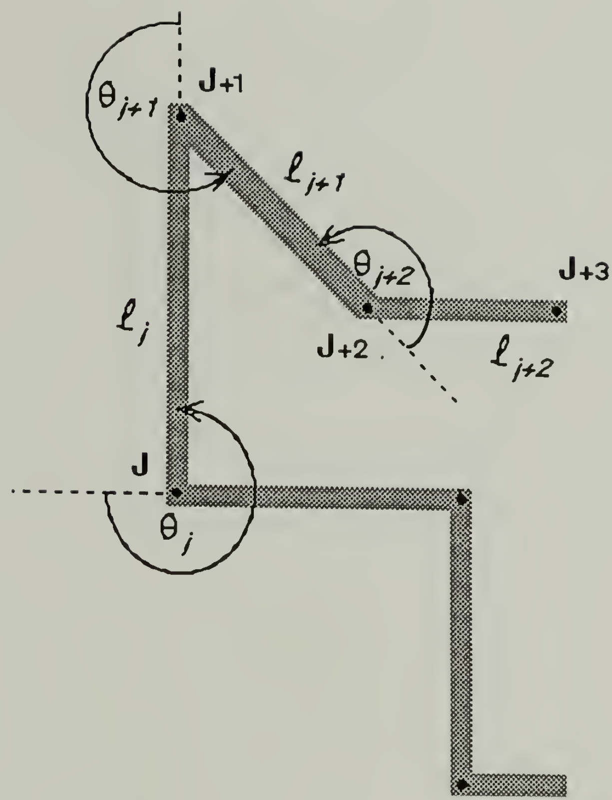

Figure 2: Conventions used for defining joint angles and computing forward kinematics.

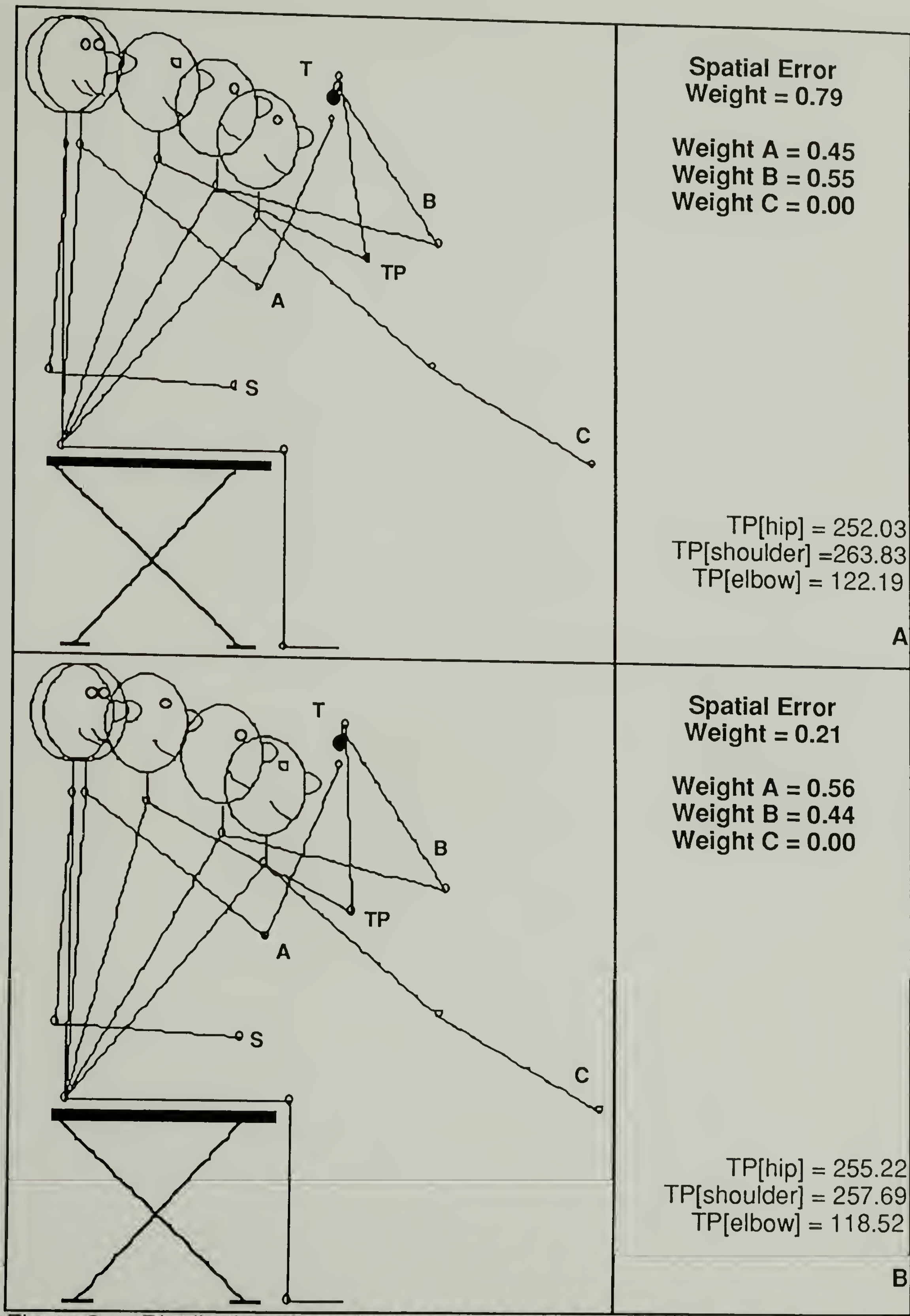

Figure 3: Distribution of weights based on task demands.

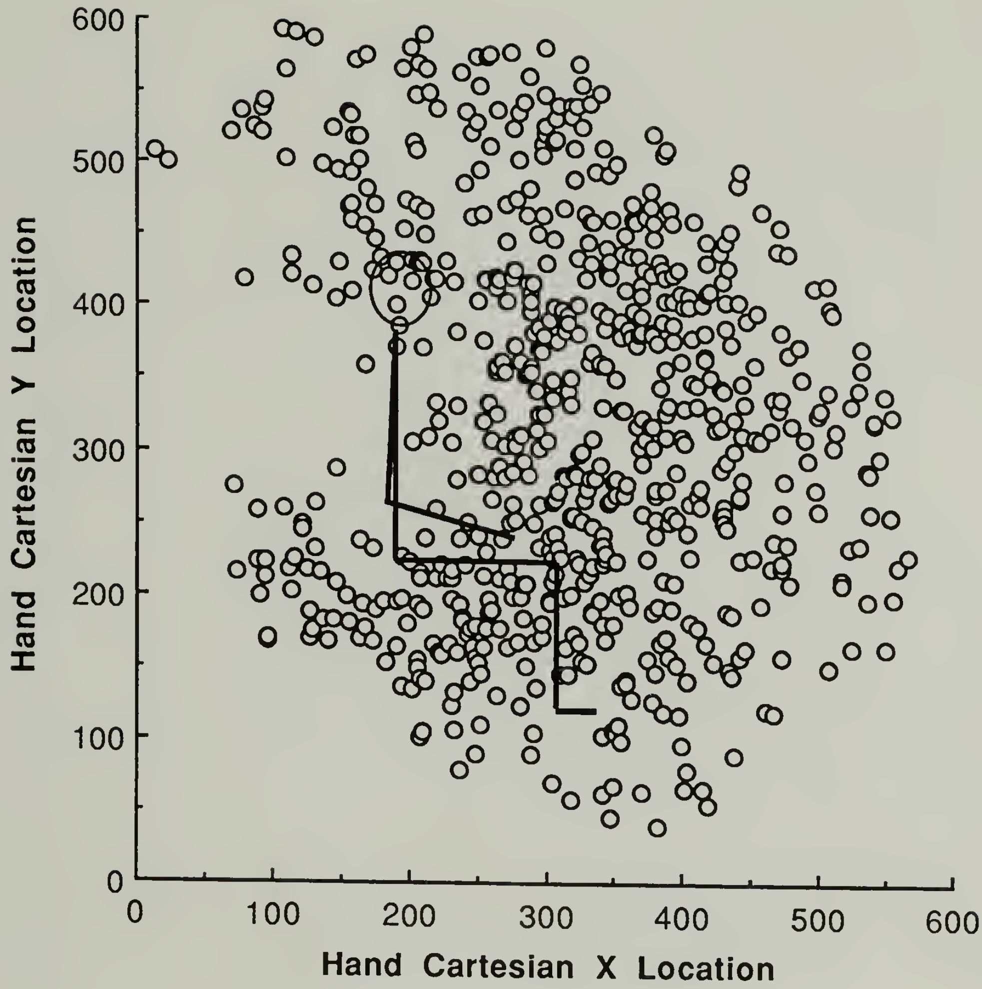

Figure 4: Hand location distribution of 600 randomly generated stored postures in the knowledge-base.

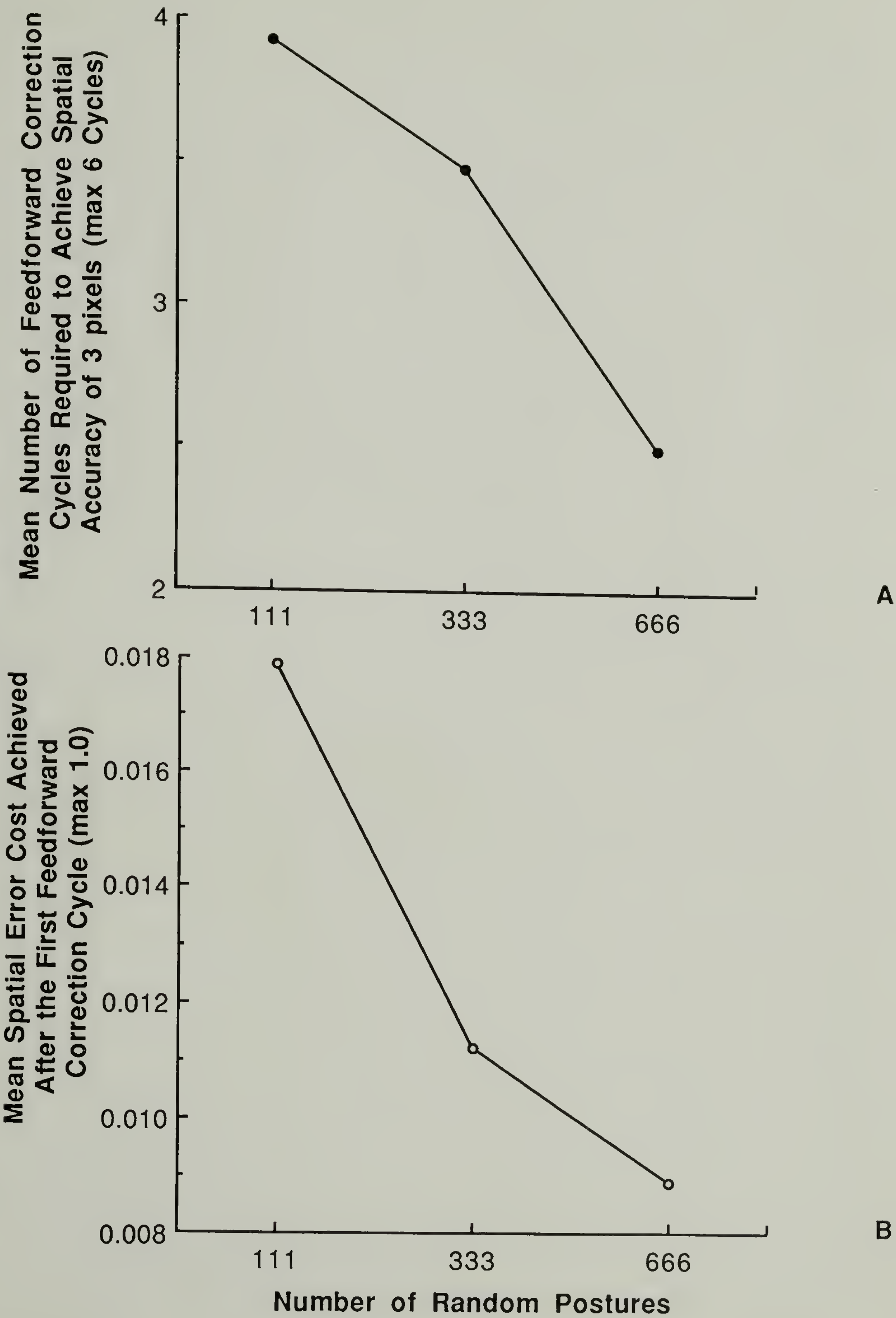

Figure 5: Effects of the number of stored postures on planning time and spatial accuracy.

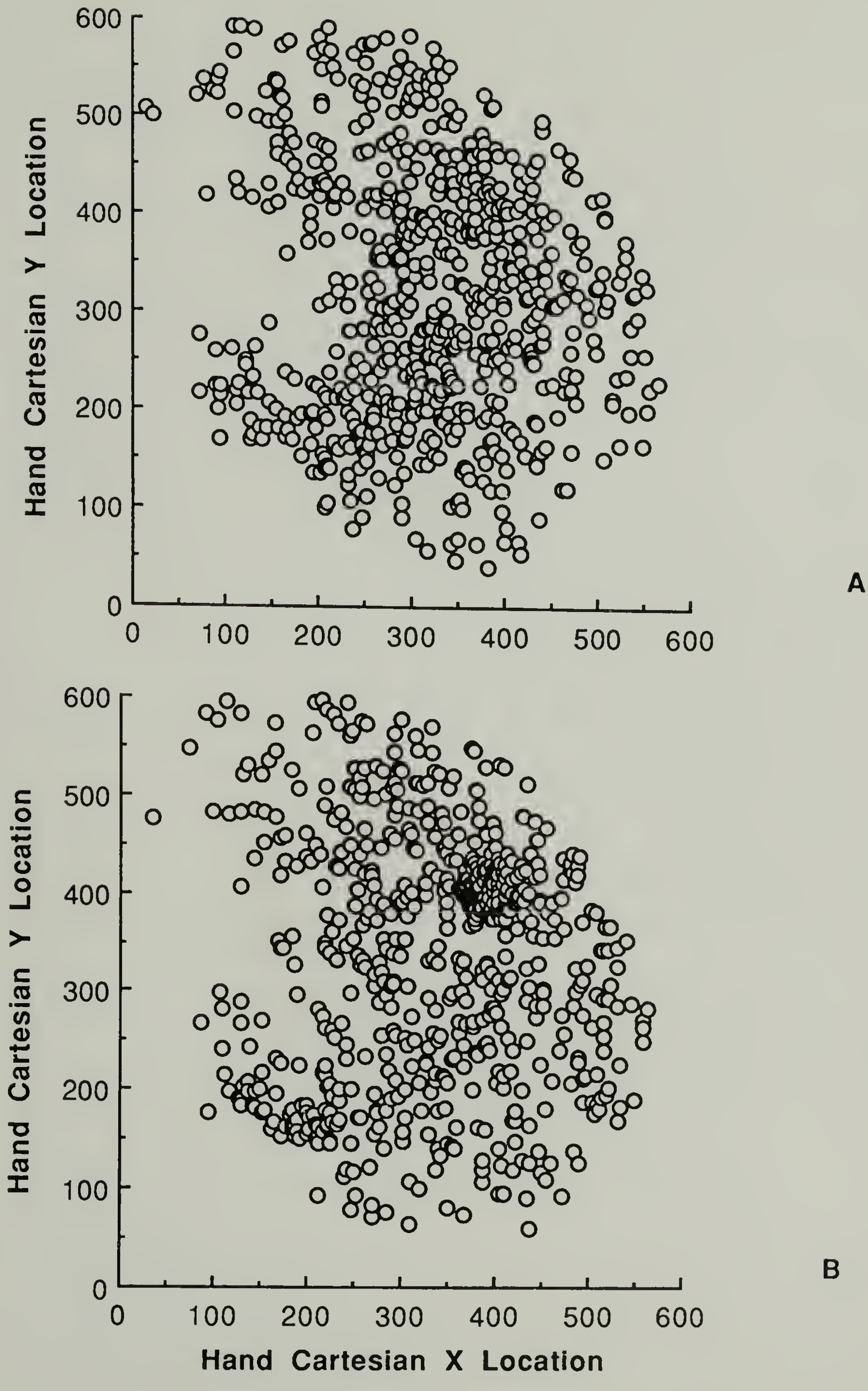

Figure 6: The development of expertise.

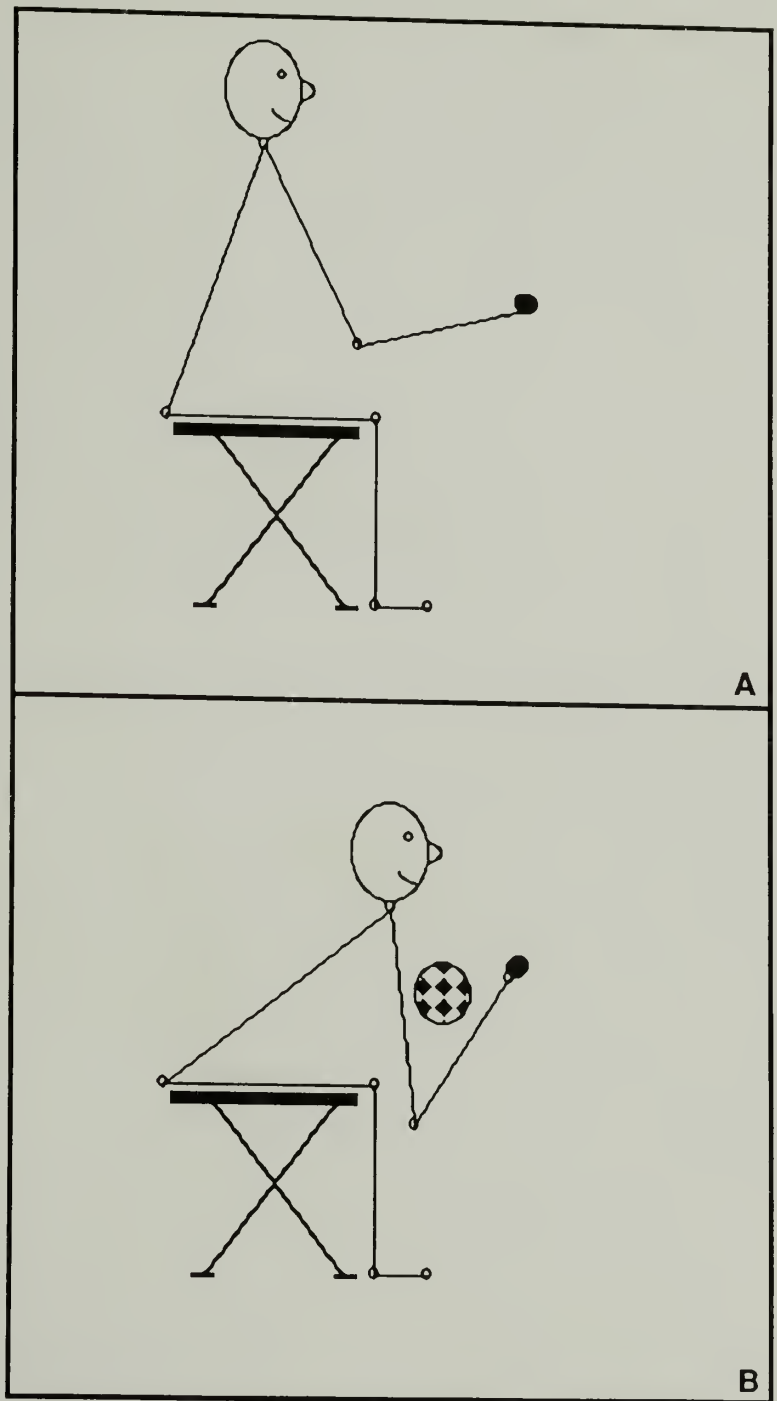

Figure 7: The degrees of freedom problem in the context of obstacle avoidance.

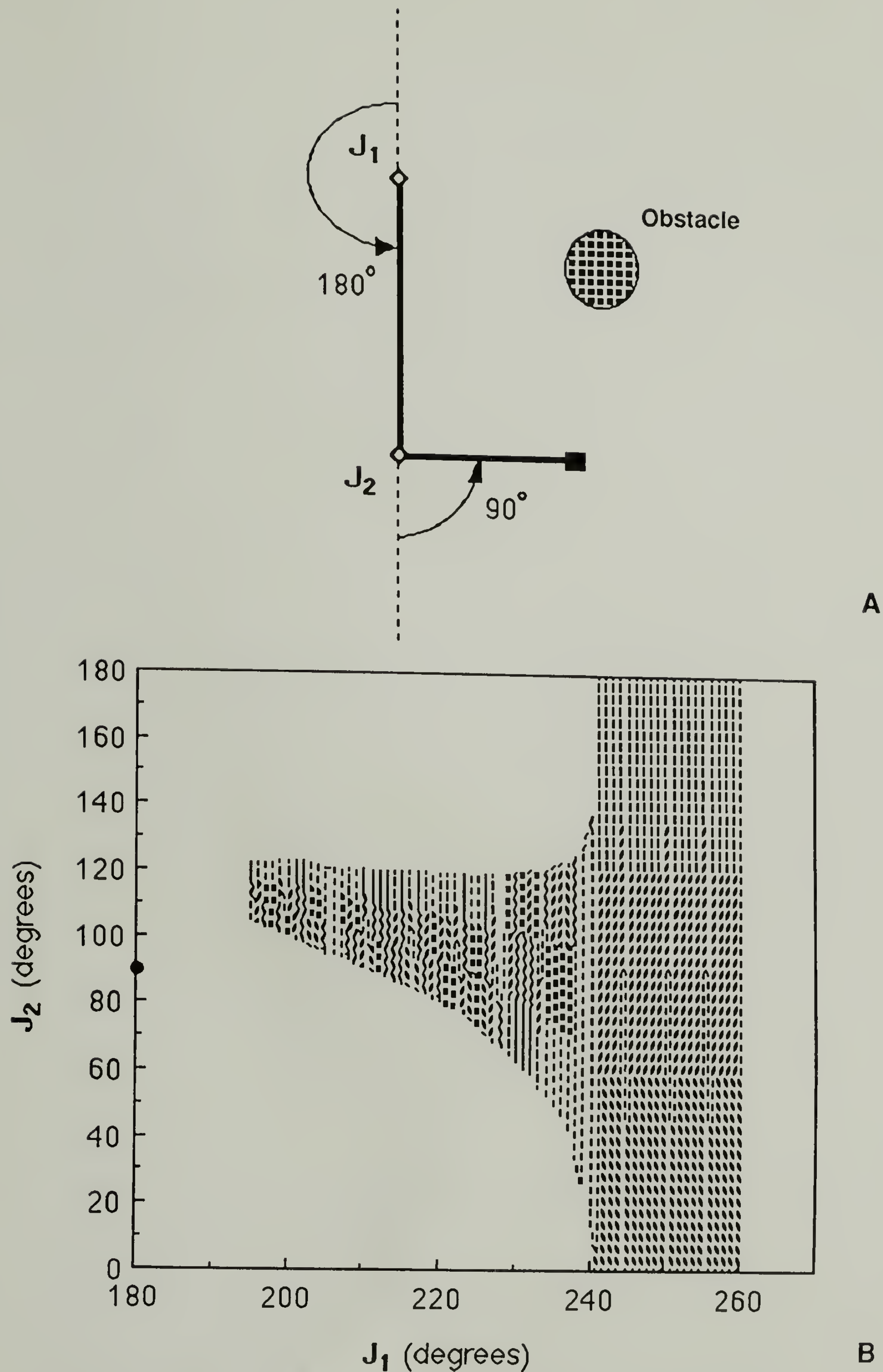

Figure 8: Configuration-space.

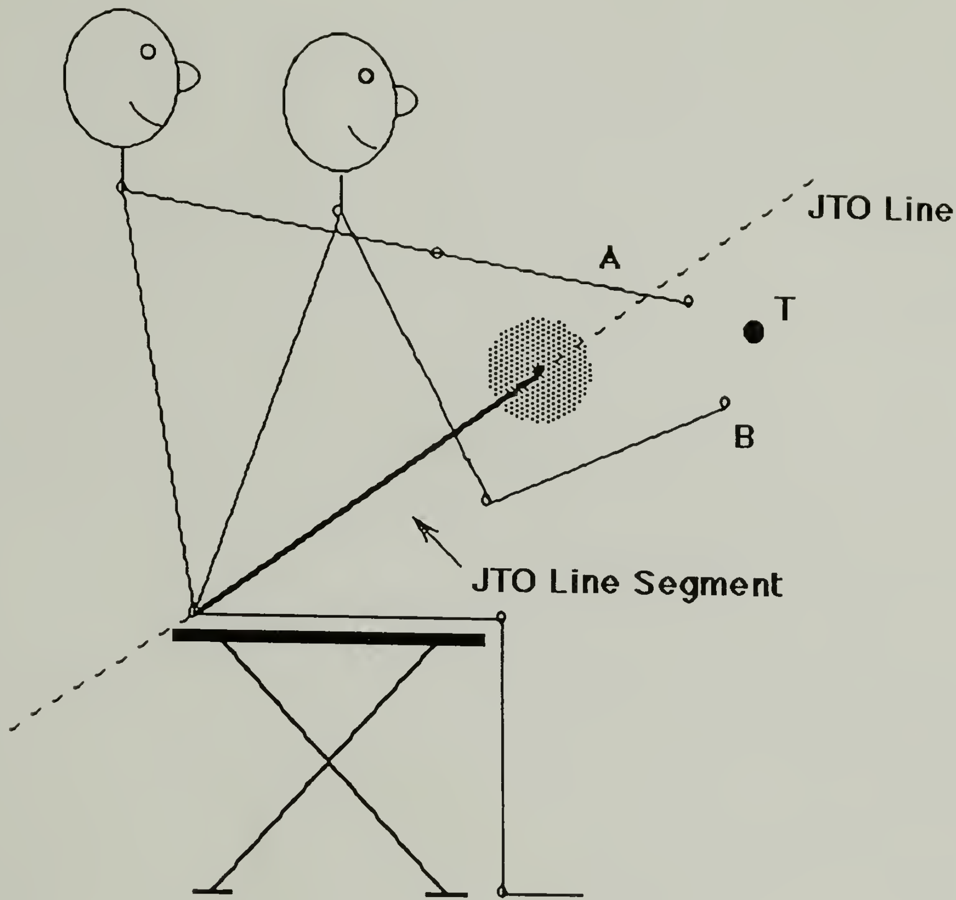

Figure 9: Partitioning of the work-space.

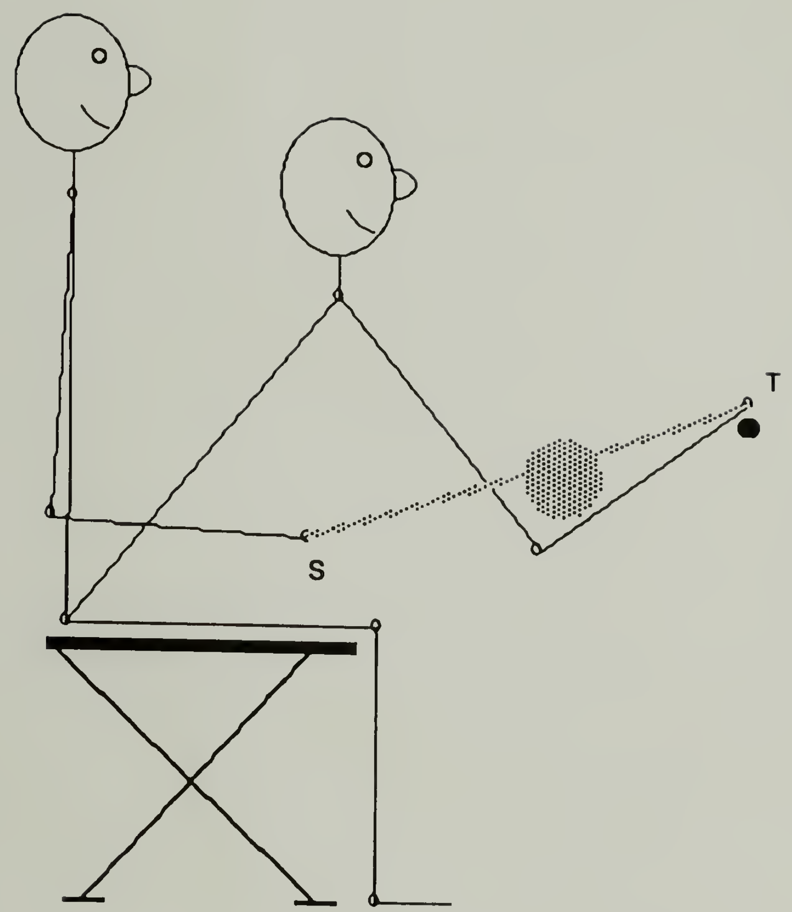

Figure 10: Trajectory leading to collision with the obstacle.

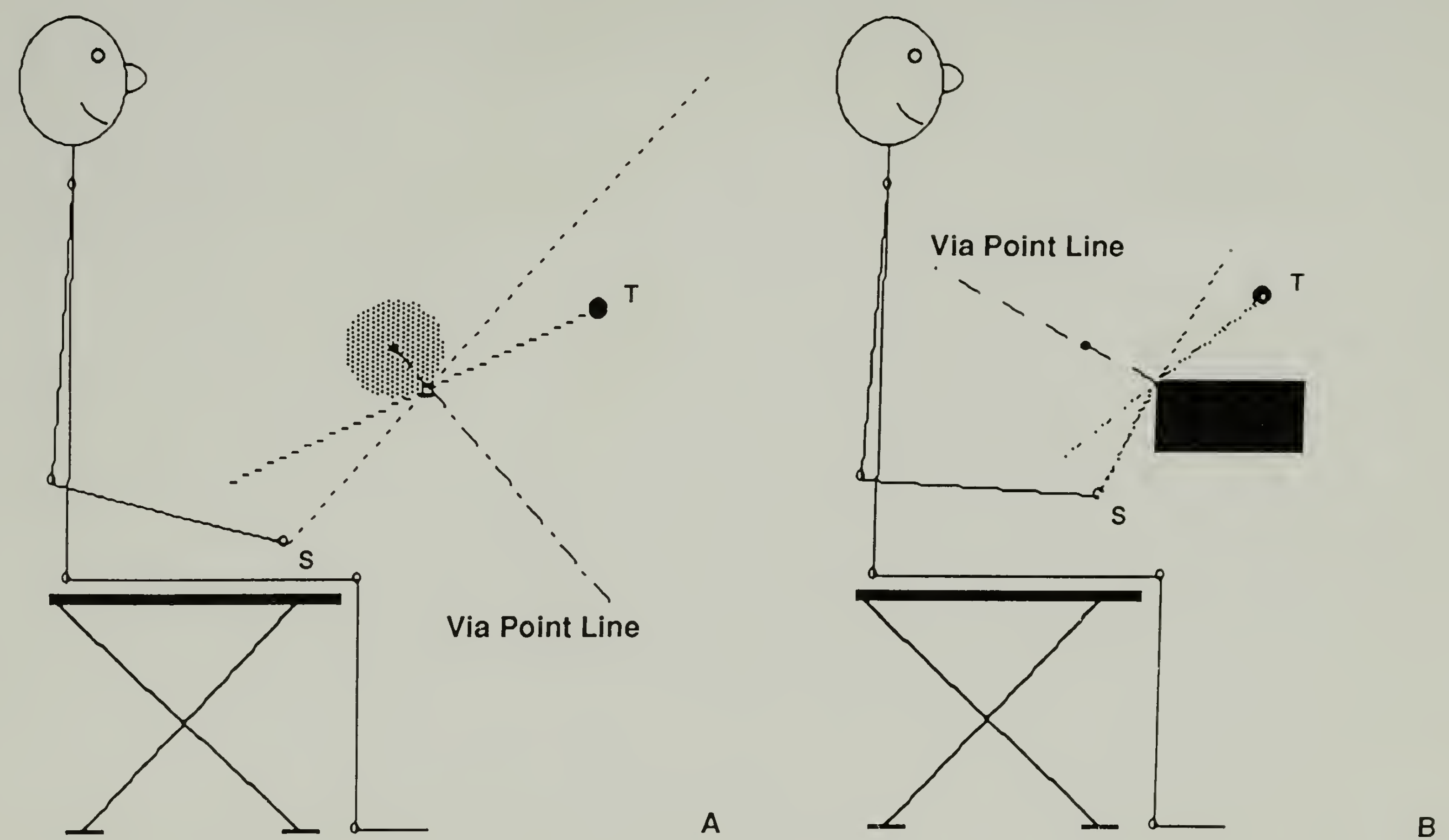

Figure 11: Method for determining the Via Point Line.

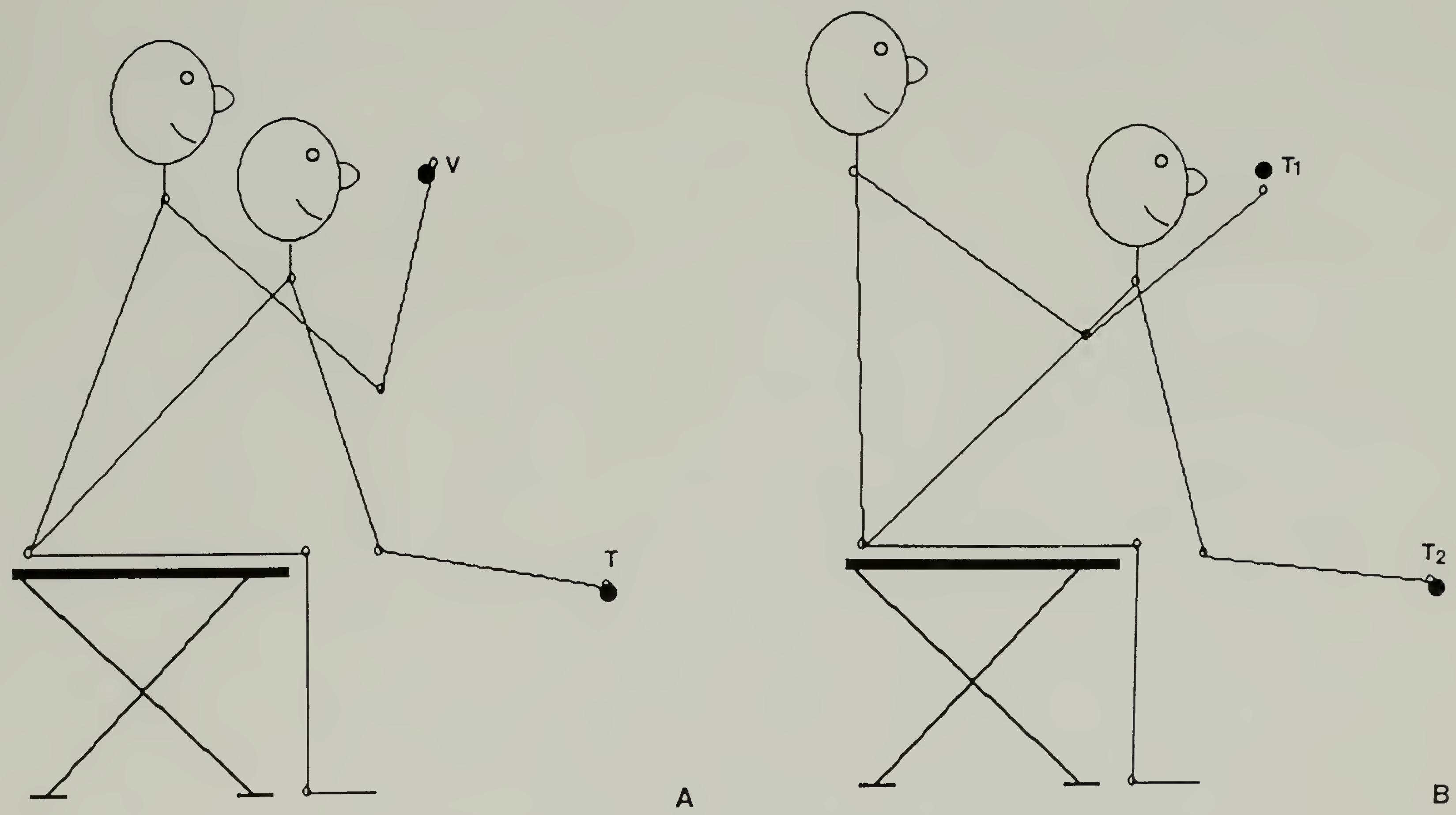

Figure 12: Anticipation effects in sequential reaching.

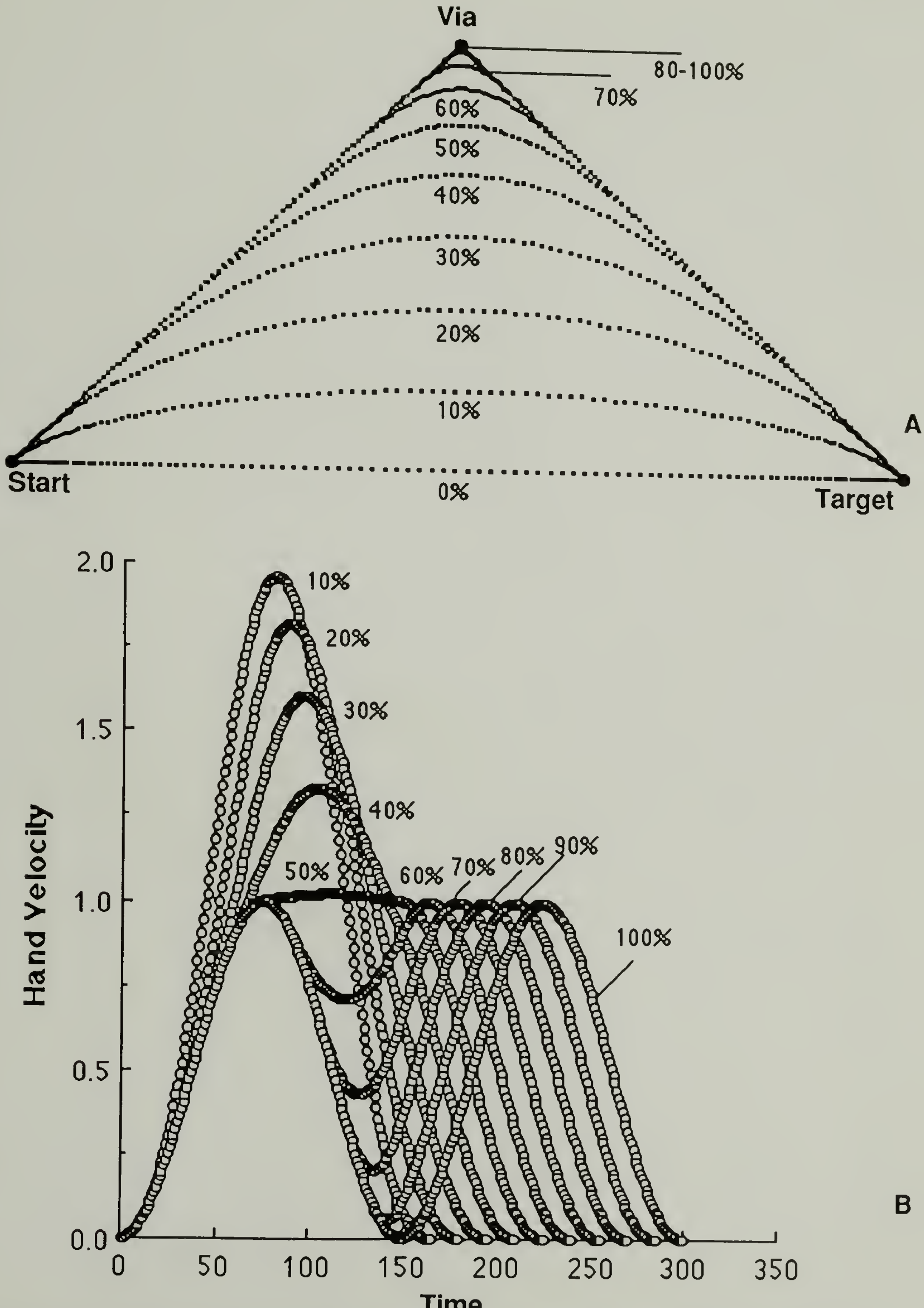

Figure 13: Effect of varying the degree of temporal overlap of two movement components on hand path and velocity profile.

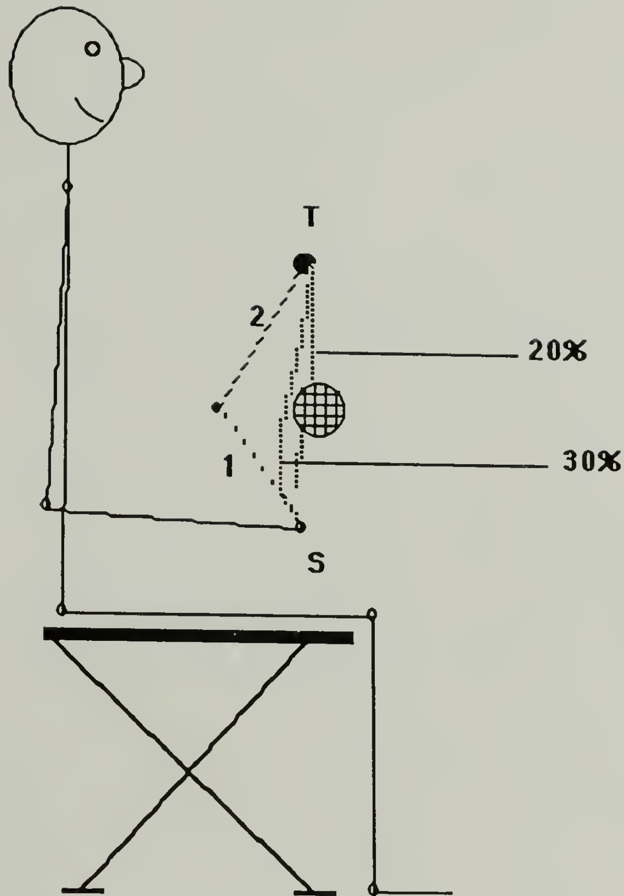

Figure 14: Changing the overlap can lead to successful obstacle avoidance.

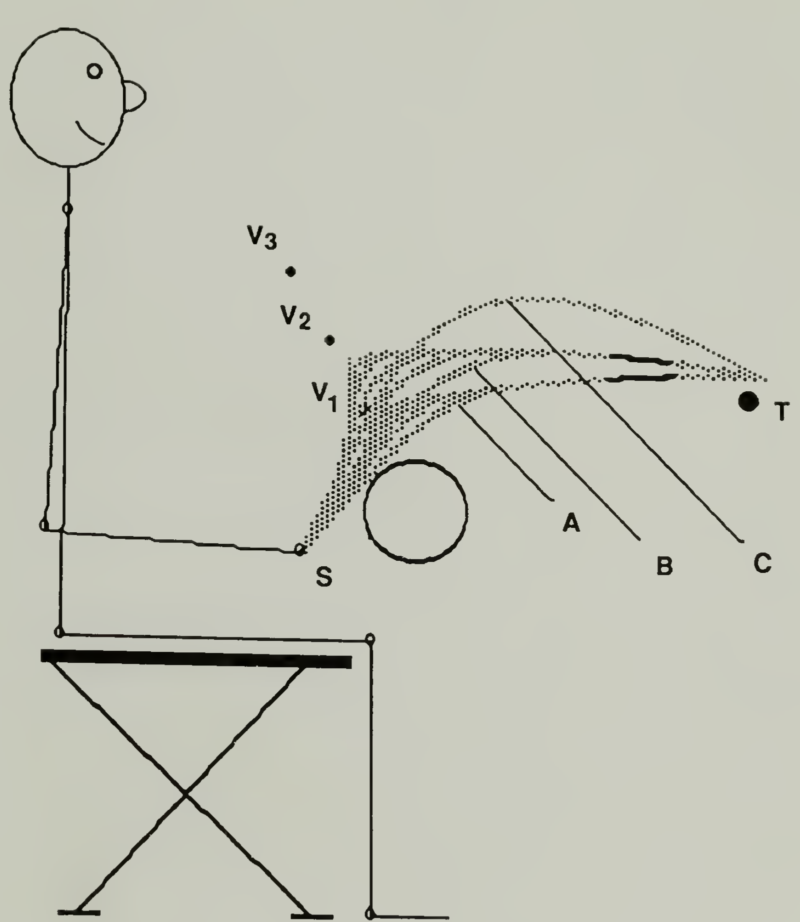

Figure 15: Moving the via point away from the obstacle can lead to successful obstacle avoidance.

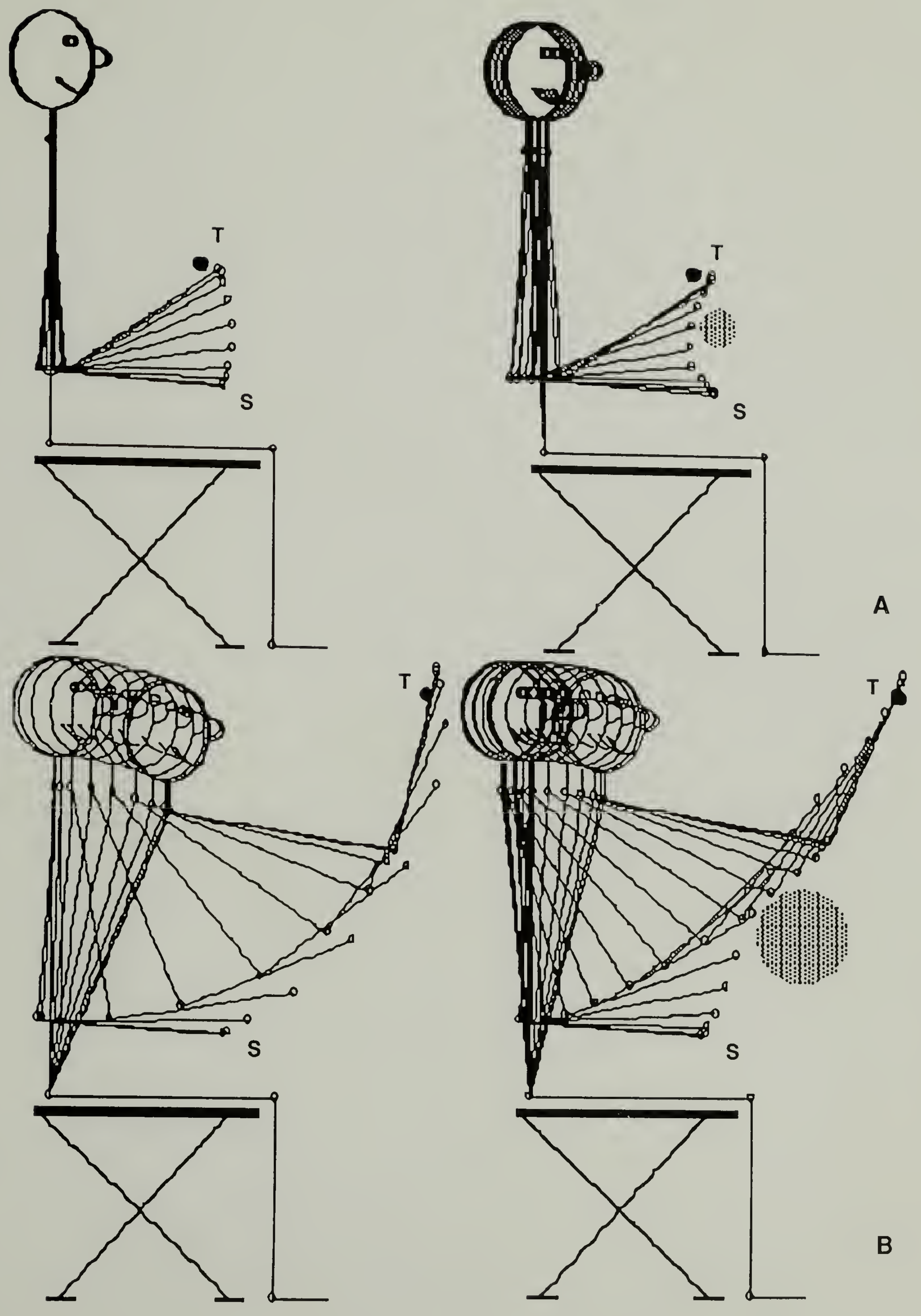

Figure 16: Avoiding a small and a large obstacle.

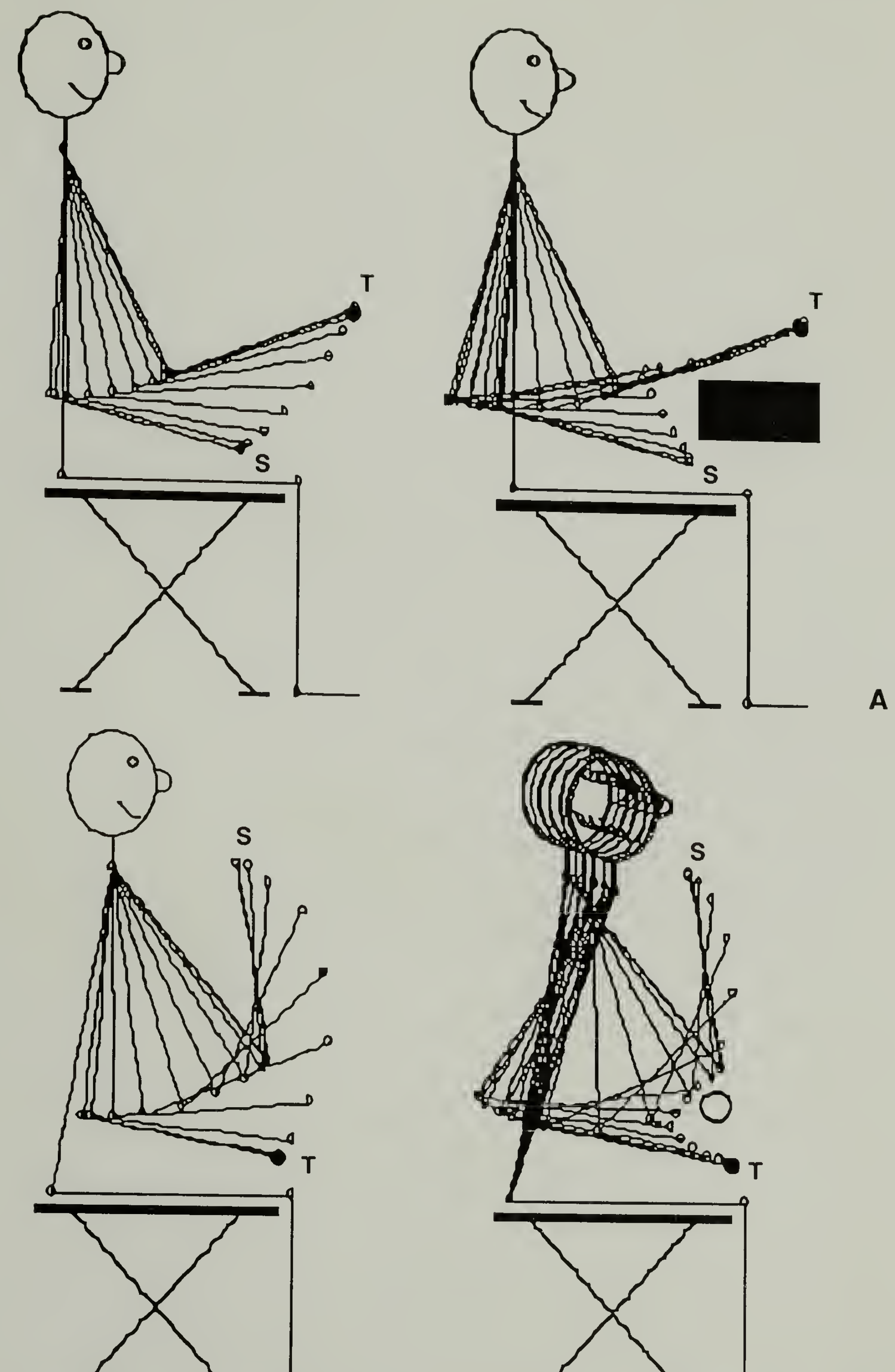

Figure 17: Avoiding a rectangular and a circular obstacle.

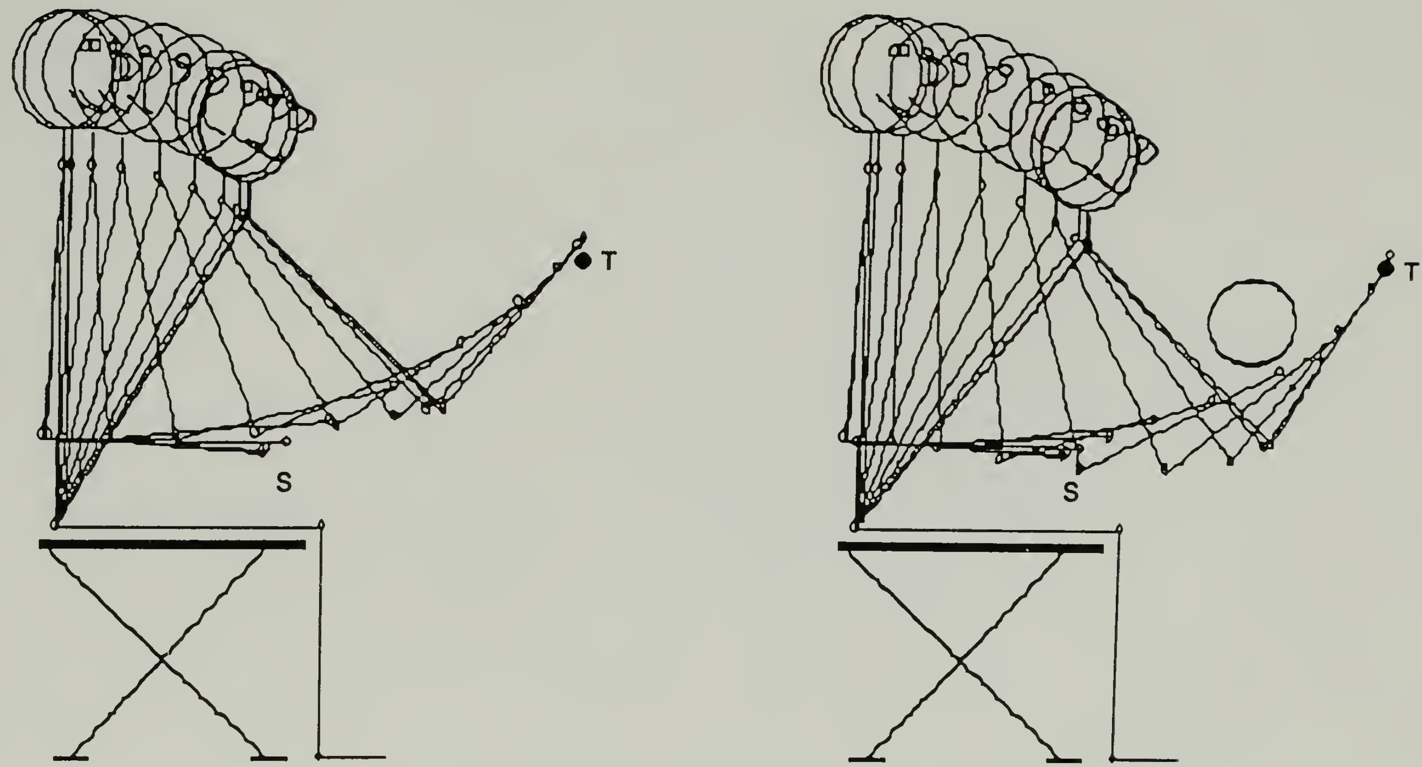

Figure 18: Reaching under a circular obstacle.

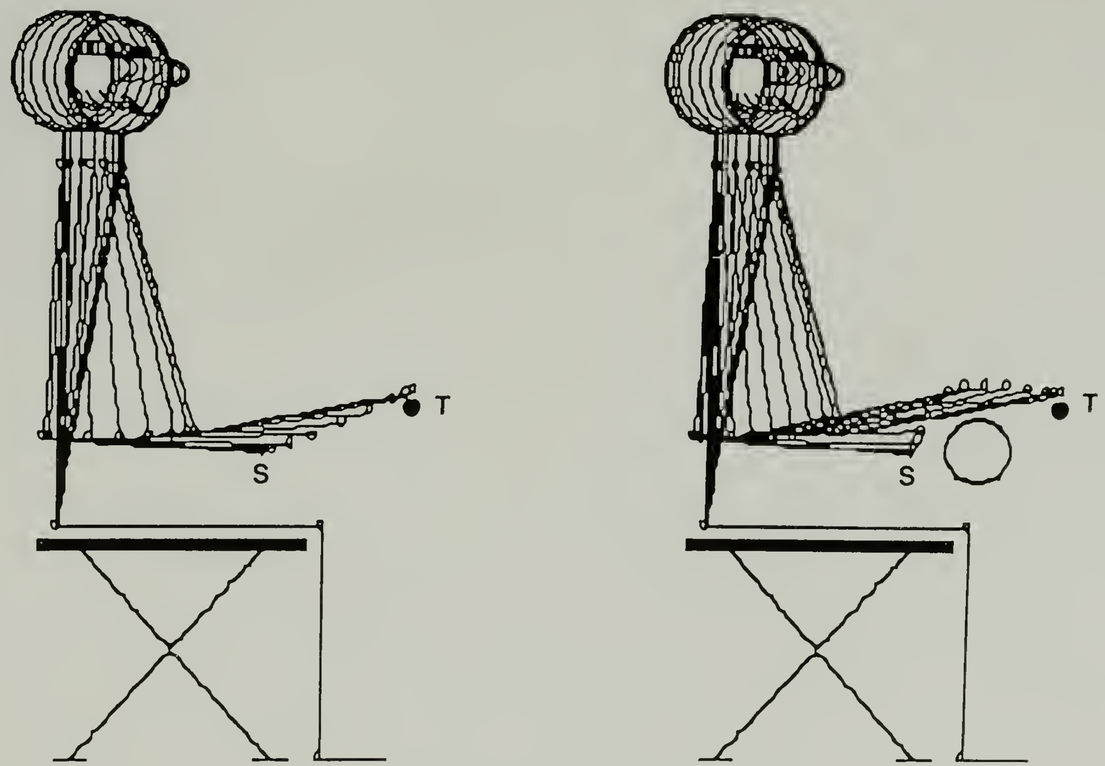

Figure 19: Reaching over a circular obstacle.

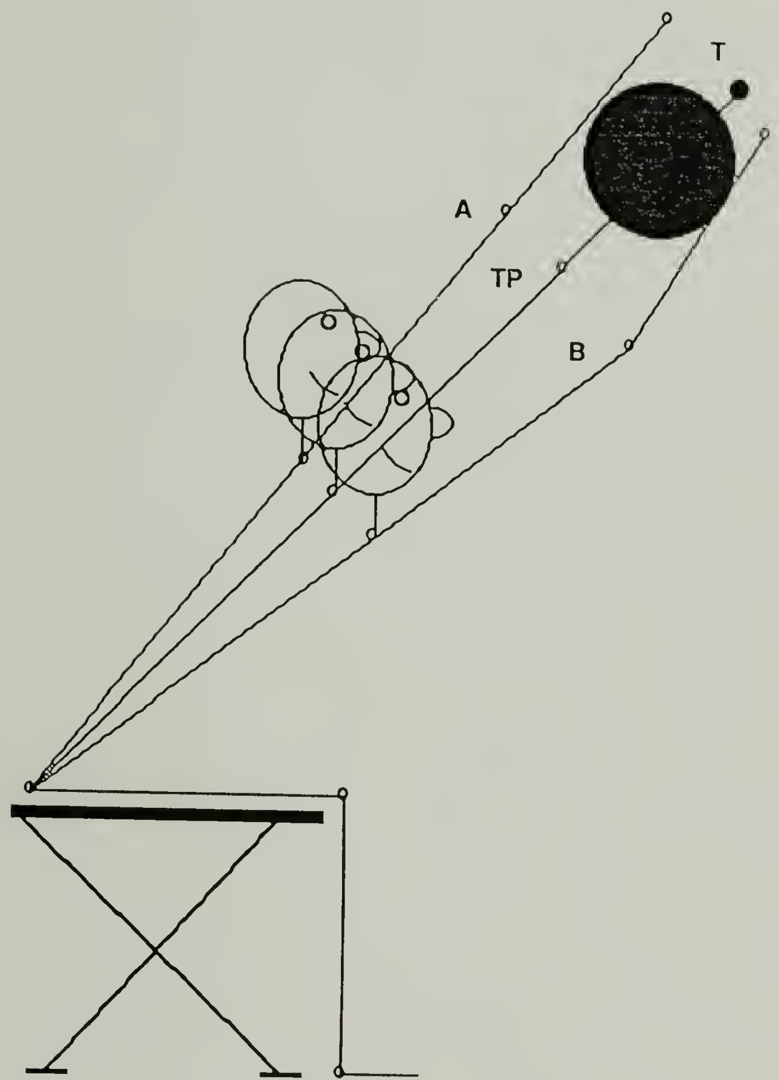

Figure 20: Failure to avoid an obstacle.

Footnotes

1. In reality, each joint has more than one degree of freedom. The shoulder, for example, has at least three mechanical degrees of freedom since it can flex/extend, abduct/adduct, and laterally/medially rotate the arm. Currently, simulations address movements in 2-dimensional space and the latter two planes are, therefore, ignored.

2. Given that total costs are non-negative numbers, only the positive side of the Gaussian is of relevance.

3. Weighted summing is a biologically plausible method which appears to be widely used in the nervous system in the form of neural population coding (Georgopoulos, 1990)

4. This result has primarily been shown for movements involving straight hand paths. For more complex, curved hand paths, the resulting speed profiles are shown to be more asymmetrical, sometimes even bimodal (Flash & Hogan, 1985)

5. Harmonic functions are solutions to Laplace's equation:

$$\sum_{i=0}^{n} \frac{\partial f(x_i^2)}{\partial x_i^2} = 0$$

where n denotes the number of variables for the function f, and each x_i is a function variable.

6. The objective function is given by:

$$C_T = \frac{1}{2} \int \sum_{i=1}^{m} \left(\frac{dz_i}{dt}\right)^2 dt$$

where z_i is the torque fed to the ith of m actuators.

7. A similar method is used to check for collisions with a circular obstacle. In this case, the equation describing every point (x, y) on the circle is:

$$x^2 + y^2 = R^2$$

where R is the radius of the circle. Once again, it is possible to solve for α and β defining the line representing each limb in its orientation prescribed by a posture under evaluation, and thus check for collision.

8. The center of the obstacle in this algorithm is assumed to be given. There do exist, however, simple heuristics to derive it geometrically, but they have not been incorporated here. One example is to use the trajectory computed by the Movement Execution component. If one were to mentally execute the trajectory and assess its collision, the obstacle center would be the hand location on the midway between the first and last interference of a trajectory component (posture) with the obstacle (Myers & Agin, 1982).

BIBLIOGRAPHY

Abend, W., Bizzi, E., & Morasso, P. (1982). Human arm trajectory formation. *Brain, 105,* 331-348.

Asatryan, D. G. & Feldman, A. G. (1965). Functional tuning of the nervous system with control of movement or maintenance of a steady posture. 1. Mechanographic analysis of the work of the joint on execution of a postural task. *Biophysics, 10,* 925-935.

Atkeson, C. G. & Hollerbach, J. M. (1985). Kinematic features of unrestrained arm movements. *The Journal of Neuroscience, 5,* 2318-2330.

Bernstein, N. (1967). *The coordination and regulation of movements.* London: Pergamon.

Berthier, N. E., Singh, S. P., Barto, A. G., & Houk, J. C. (1993). Distributed representations of limb motor programs in arrays of adjustable pattern generators. *Journal of Cognitive Neuroscience, 5,* 56-78.

Bizzi, E. & Mussa-Ivaldi, F. A. (1989). Geometrical and mechanical issues in movement planning and control. In M. I. Posner (Ed.), *Handbook of Cognitive Science* (pp. 769-792). Cambridge, MA: MIT Press.

Bizzi, E. (1980). Central and peripheral mechanisms in motor control. In G. E. Stelmach & J. Requin (Eds.), *Tutorials in Motor Behavior* (pp. 131-144). Amsterdam: North-Holland.

Bizzi, E., Accornero, N., Chapple, W., & Hogan, W. (1984). Posture control and trajectory formation during arm movement. *Journal of Neuroscience, 4,* 2738-2744.

Bizzi, E., Hogan, N., Mussa-Ivaldi, F. A., & Giszter, S. (1992). Does the nervous system use equilibrium-point control to guide single and multiple joint movements? *Behavioral and Brain Sciences, 15,* 603-613.

Brüwer, M. & Dean, J. (1993). *Generation of arm trajectories in a two-dimensional workspace with obstacles. I. Experimental results.* (submitted manuscript).

Bullock, D. & Grossberg, S. (1988). Neural dynamics of planned arm movements: Emergent invariants and speed-accuracy properties during trajectory formation. *Psychological Review, 95,* 49-90.

Bullock, D., Grossberg, S., & Guenther, F. H. (1993). A self-organizing neural model of motor equivalent reaching and tool use by a multijoint arm. *Journal of Cognitive Neuroscience, 5,* 408-435.

Connolly, C. I. & Burns, J. B. (1992). *The basal ganglia and the planning of actions.* Technical Report, Computer Science Department, University of Massachusetts (Amherst).

Connolly, C. I., & Burns, J. B. (1993). A new striatal model and its relationship to basal ganglia diseases. *Neuroscience Research, 16,* 271-274.

Connolly, C. I., & Grupen, R. A. (1993). The applications of harmonic functions to robotics. *Journal of Robotic Systems, 10* (7), 931-946.

Crossman, E. R. F. W. & Goodeve, P. J. (1963/1983). Feedback control of hand-movement and Fitts' Law. *Quarterly Journal of Experimental Psychology, 35A,* 251-278.

Cruse, H., Brüwer, M., & Dean, J. (1993). Control of three- and four-joint arm movement: Strategies for a manipulator with redundant degrees of freedom. *Journal of Motor Behavior 25,* 131-139.

Dean, J. & Brüwer, M. (1992). *Control of human arm movements in 2-dimensions: paths and joint control in avoiding simple linear obstacles.* (unpublished manuscript).

Diamond, A. (1990). Developmental time course in human infants and infant monkeys, and the neural bases of, inhibitory control in reaching. *Annals of the New York Academy of Sciences, 608,* 637-676.

Diamond, A. (1993). Neurophysiological insights into the meaning of object concept development. In M. H. Johnson (Ed.), *Brain Development and Cognition: A Reader* (pp. 208-247). Cambridge, MA: Blackwell Publishers.

Engelbrecht, S. E. & Rosenbaum, D. A. (1993). *A computational model of optimal movement selection*. (submitted manuscript).

Feldman, A. G. (1966). Functional tuning of the nervous system with control of movement or maintenance of a steady posture: II. Controllable parameters of the muscles. *Biophysics, 11*, 565-578.

Fischer, M. H., Rosenbaum, D. A., Loukopoulos, L. D., & Szymkowiak, A. (1993). *Speed and sequential effects in reaching*. (submitted manuscript).

Flash, T. & Henis, E. (1991). Arm trajectory modification during reaching towards visual targets. *Journal of Cognitive Neuroscience, 3*, 220-230.

Flash, T. & Hogan, N. (1985). The coordination of arm movements: An experimentally confirmed mathematical model. *The Journal of Neuroscience, 5*, 1688-1703.

Flash, T., Henis, E., Inzelberg, R., & Korczyn, A. D. (1992). Timing and sequencing of human arm trajectories: Normal and abnormal motor behaviour. *Human Mov ement Science, 11*, 83-100.

Gentner, D. R., Grudin, J., & Conway, E. (1980). *Finger movements in transcription typing*. La Jolla, CA: Center for Human Information Processing, University of California, San Diego.

Georgopoulos, A. P. (1990). Neurophysiology of reaching. In M. Jeannerod (Ed.), *Attention and Performance XIII* (pp. 227-263). Hillsdale, NJ: Erlbaum.

Georgopoulos, A. P. (1991). Higher order motor control. *Annual Review of Neuroscience, 14*, 361-377.

Gibson, J. J. (1958). Visually controlled locomotion and visual orientation in animals. *British Journal of Psychology, 49*, 182-194.

Gibson, J. J. (1977). The theory of affordances. In R. E. Shaw and J. Bransford (Eds.), *Perceiving, acting, and knowing: Toward an ecological psychology* (pp. 67-82). Hillsdale, NJ: Erlbaum.

Gibson, J. J. (1979). *The ecological approach to visual perception.* Boston: Houghton-Mifflin.

Henis, E. & Flash, T. (1992). A computational mechanism to account for averaged modified hand trajectories. In J. E. Moody, S. J. Hansen, R. P. Lippmann, R. P. (Eds.), *Neural Information Processing Systems, 4,* (pp. 619-626). San Mateo, CA: Morgan Kaufmann.

Kaminski, T. & Gentile, A. M. (1986). Joint control strategies and hand trajectories in multijoint pointing movements. *Journal of Motor Behavior, 18,* 261-278.

Kawato, M., Maeda, Y., Uno, Y., & Suzuki, R. (1990). Trajectory formation of arm movement by cascade neural network model based on minimum torque-change criterion. *Biological Cybernetics, 62,* 275-288.

Keele, S. W. (1968). Movement control in skilled motor performance. *Psychological Bulletin, 70,* 387-403.

Kelso, J. A. S. & Holt, K. G. (1980). Exploring a vibratory systems analysis of human movement production. *Journal of Neurophysiology, 43,* 1183-1196.

Loukopoulos, L. D., Rosenbaum, D. A., Meulenbroek, R. G. J., Vaughan, J. (1993). Computations for obstacle avoidance. Paper presented at the *34th Annual Meeting of the Psychonomics Society,* Washington, D. C., November 5-7.

Lozano-Perez, T. (1983). Spatial planning: A configuration space approach. *IEEE Transactions on Computers, C-32 (2),* 108-120.

Mark, L. S., Baillett, J. A., Craver, K. D., Douglas, S. D., & Fox, T. (1990). What an actor must do in order to perceive the affordance of sitting. *Ecological Psychology, 2,* 325-366.

Meulenbroek, R. G. J., Rosenbaum, D. A., Thomassen, A. J. W. M., & Loukopoulos, L. D. (1993). Applying a reaching control model to drawing behavior. Proceedings of the*Sixth International Conference on Handwriting and Drawing* (pp. 10-12). Paris: Telecom.

Meulenbroek, R. G. J., Rosenbaum, D. A., Thomassen, A. G., & Loukopoulos, L. D. (in press). A model of limb segment coordination in drawing behavior. In C. Faure, P. Keuss, G. Lorette, A. Vinter (Eds.), *Advances in handwriting and drawing: A multi-disciplinary approach..* Paris, France: Telecom.

Meulenbroek, R. G. J., Rosenbaum, D. A., Thomassen, A. J., & Schomaker, L. R. (1993). Limb-segment selection in drawing behavior. *Quarterly Journal of Experimental Psychology, 46A (2)*, 273-299.

Minieka, E. (1978). *Optimization algorithms for networks and graphs.* Marcel Dekker.

Moll, K. L., & Danilloff, R. G. (1971). Investigation of the timing of velar movements during speech. *Journal of the Acoustical Society of America, 50*, 678-684.

Morasso, P. (1981). Spatial control of arm movements. *Experimental Brain Research, 42*, 223-227.

Muthuswamy, S. & Manoochehri, S. (1992). Optimal path planning for robot manipulators. *Transactions of the ASME, 114*, 586-594.

Myers, J. K., & Agin, G. J. (1982). A supervisory collision-avoidance system for robot controllers. In W. J. Book (Ed.), *Robotics research and advanced applications.* (pp. 225-232). New York, NY: The American Society of Mechanical Engineers.

Newell, A. M. & Rosenbloom, P. S. (1981). Mechanisms of skill acquisition and the law of practice. In J. R. Anderson (Ed.), *Cognitive skills and their acquisition* (pp. 1-55). Hillsdale, NJ: Erlbaum.

Pufall, P., & Dunbar, C. (1992). Perceiving whether or not the world affords stepping onto and over: A developmental study. *Ecological Psychology, 4(1)*, 17-38.

Rosenbaum, D. A. (1991). *Human motor control.* San Diego: Academic Press.

Rosenbaum, D. A., Engelbrecht, S. E., Bushe, M. M., & Loukopoulos, L. D. (1993a). A model for reaching control. *Acta Psychologica, 82*, 237-250.

Rosenbaum, D. A., Engelbrecht, S. E., Bushe, M. M., & Loukopoulos, L. D. (1993b). Knowledge model for selecting and producing reaching movements. *Journal of Motor Behavior, 25,* 217-227.

Rosenbaum, D. A. & Krist, H. (in press). Antecedents of action. In H. Heuer & S. W. Keele (Eds.), *Handbook of Motor Control*. Academic Press.

Rosenbaum, D. A., Loukopoulos, L. D., Meulenbroek, R. G. J., Vaughan, J., & Engelbrecht, S. E. (1993c). *Planning reaches by evaluating stored postures.* (to appear in *Psychological Review*).

Rosenbaum, D. A., Marchak, F., Barnes, H. J., Vaughan, J., Slotta, J., & Jorgensen, M. (1990). Constraints for action selection: Overhand versus underhand grips. In M. Jeannerod (Ed.), *Attention and Performance XIII* (pp. 321-342). Hillsdale, NJ: Lawrence Erlbaum Associates.

Rosenbaum, D. A., Slotta, J. D., Vaughan, J., & Plamondon, R. J. (1991). Optimal movement selection. *Psychological Science, 2,* 86-91.

Rosenbaum, D. A., Vaughan, J., Barnes, H. J., & Jorgensen, M. J. (1992). Time course of movement planning: Selection of hand grips for object manipulation. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 18,* 1058-1073.

Rosenbaum, D. A., Vaughan, J., Jorgensen, M. J., Barnes, H. J., & Stewart, E. (1993d). Plans for object manipulation. In D. E. Meyer & S. Kornblum (Eds.), *Attention and Performance XIV — A silver jubilee: Synergies in experimental psychology, artificial intelligence and cognitive neuroscience* (pp. 803-820). Cambridge: MIT Press, Bradford Books.

Shadmehr, R. (1993). Control of equilibrium position and stiffness through postural modules. *Journal of Motor Behavior, 25,* 228-241.

Soechting, J. F., & Lacquaniti, F. (1981). Invariant characteristics of a pointing movement in man. *Journal of Neuroscience, 1,* 710-720.

Turvey, M. T. (1990a). Coordination. *American Psychologist, 45, 8,* 938-953.

Turvey, M. T. (1990b). The challenge of a physical account of action: A personal view. In H. T. A. Whiting, O. G. Meijer, & P. C. van Wieringen (Eds.), *The natural-physical approach to movement control* (pp. 57-93). Amsterdam: VU University Press.

Udupa, S. M. (1977). Collision detection and avoidance in computer controlled manipulators. *Proceedings of the 5th International Joint Conference in Artificial Intelligence*, Massachusetts Institute of Technology.

Uno, Y., Kawato, M., & Suzuki, R. (1989). Formation and control of optimal trajectory in human multijoint arm movement: Minimum torque-change model. *Biological Cybernetics, 61*, 89-101.

Vaughan, J., Rosenbaum, D. A., Loukopoulos, L. D., & Engelbrecht, S. E. (1993a). *Finding final postures* (manuscript in preparation).

Vaughan, J., Rosenbaum, D. A., Moore, C., & Diedrich, F. (1993b). *Cooperative selection of movements: The optimal selection model.* (submitted manuscript).

Warren, W. H. & Whang, S. (1987). Visual guidance of walking through apertures: Body-scaled information for affordances. *Journal of Experimental Psychology, 13*, 371-383.

Warren, W. H. Jr. (1984). Perceiving affordances: Visual guidance of stair climbing. *Journal of Experimental Psychology: Human Perception and Performance, 10*, 683-703.

Woodworth, R. S. (1899). The accuracy of voluntary movement. *Psychological Review Monograph Supplements, 3*, Whole No. 3.