Biol. Cybern. 73, 569–575 (1995)

Biological
Cybernetics
© Springer-Verlag 1995

Path integration – a network model

Thomas Wittmann, Helmut Schwegler

Zentrum für Kognitionswissenschaften, Institut für Theoretische Physik, Universität Bremen, D-28334 Bremen, Germany

Received: 25 April 1995 / Accepted in revised form: 30 May 1995

Abstract. Path integration is a primary means of navigation for a number of animals. We present a model which performs path integration with a neural network. This model is based on a neural structure called a *sinusoidal array*, which allows an efficient representation of vector information with neurons. We show that exact path integration can easily be achieved by a neural network. Thus deviations from the direct home trajectory, found previously in experiments with ants, can not be explained by computational limitations of the nervous system. Instead we suggest that the observed deviations are caused by a strategy to simplify landmark navigation.

1 Introduction

A number of animals use path integration (dead reckoning) as a primary means of navigation. Experiments indicate that it is used by a number of animal groups including spiders (Görner and Claas 1985; Moller and Görner 1994), mammals (Mittelstaedt and Mittelstaedt 1982; Mittelstaedt and Glasauer 1991; Seguinot et al. 1993), and ants (Wehner and Srinivasan 1981; Müller and Wehner 1988).

Although this calculation is subject to accumulation of errors, it provides the animal with an estimation of its current position relative to its nest that can be refined by landmark navigation.

In order to navigate with path integration the animal needs a way to determine its orientation relative to a fixed direction – a compass (Wehner 1994) – and some means to compute distances (Schäfer and Wehner 1993). Using this information it can compute an estimation of its current position by vectorially adding all distances traveled.

The most elaborate empirical studies on path integration have been done with the desert ant *Cataglyphis* (Wehner and Srinivasan 1981; Müller and Wehner 1988; Wehner and Müller 1993). Müller and Wehner (1988) found that these animals make systematic errors when they are forced to walk in a straight line through a narrow channel for a certain distance, then to turn about a fixed angle and to walk in a straight line again for another distance. At the end of this two-leg trajectory they did not head directly for their nest. Instead their homing direction differed from the straight line to the nest by up to 25°.

In the present paper we show how a neural network that is capable of performing path integration can be designed. Unlike the models of Hartmann and Wehner (1995) and Müller and Wehner (1988) we do not try to fit free parameters of our model to the observed errors. Instead we argue that an exact path integration is easy to achieve also with neural hardware. As the above-mentioned models could also be tuned to achieve nearly exact path integration by choosing the free parameters not to fit the experimental data but to provide the best possible performance, there seems to be no computational limitation in the ant's brain that might cause these errors. Instead there must be some advantage for the ant to start its return to the nest in a direction that differs from the direct return path. Possibly this can be explained if we consider that these animals use landmark navigation to correct inaccuracies in the position computed by path integration (Wehner 1991). We will show that these 'errors' might occur because of a simple strategy to simplify landmark navigation when the animal returns to its nest.

2 Navigation by path integration

In order to compute the return path to the nest the animal has to 'compute' incrementally the following distances and angles (Fig. 1):

$$\begin{aligned} l_{n+1} &= \sqrt{l_n^2 + (v_n \Delta t_n)^2 + 2 l_n v_n \Delta t_n \cos\delta_n} && (1) \\ &\approx l_n + v_n \Delta t_n \cos(\delta_n) \\ \phi_{n+1} &= \phi_n + \arcsin\left(\frac{v_n \Delta t_n \sin\delta_n}{l_n}\right) && (2) \\ &\approx \phi_n + \frac{v_n \Delta t_n}{l_n} \sin\delta_n \end{aligned}$$

where l_n is distance to nest in step n, ϕ_n is direction from nest to current position, v_n is average speed during step n, Δt_n is duration of step n, γ_H is course direction, and $\delta_n = \gamma_H - \phi_n$.

Correspondence to: T. Wittmann

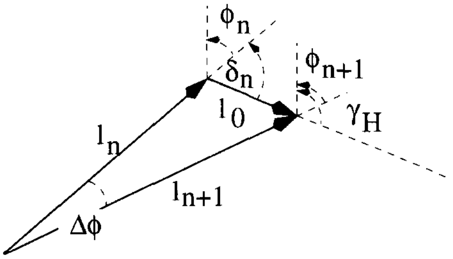

Fig. 1. Iterative computation of the distance and direction from the nest to the current position

Using vector notation this can be simplified to

$$\mathbf{R}_{n+1} = (l_{n+1}, \phi_{n+1}) = \mathbf{R}_n + \mathbf{v}_n \Delta t_n \tag{3}$$

where $\mathbf{v}_n = (v_n, \gamma_H)$.

3 The neuron model

Before we can elaborate on how the computations in (3) can be performed by a neural network we have to decide which model we will use to describe the neurons. Various models of biological neurons have been suggested with various levels of abstraction. One of the most popular models is the *average firing rate* model (see for instance Rumelhart and McClelland 1986). It abstracts from the details of spike generation and describes the average number of spikes generated by a neuron in a fixed interval of time. In this model every neuron y_j computes a weighted sum of all inputs, subtracts a threshold θ_j and applies a sigmoidal function f:

$$y_j = f(\sum_i w_{i,j}\, y_i - \theta_j) \tag{4}$$

Throughout this paper we assume that the neurons in the path integration system can be described adequately in terms of this model.

4 Representation of vectors with neurons

Two-dimensional vectors can elegantly be represented by so-called *phasors* , i.e. as amplitude and phase of a sinus wave: Every vector $\mathbf{v} = (r, \phi)$ can be encoded as a sinusoidal wave $f_{\mathbf{v}}(t) = r\ \cos(\omega t + \phi)$. This encoding has the advantage that summation of vectors can be achieved by addition of the corresponding phasors. If two vectors $\mathbf{v}_1 = (r_1, \phi_1)$ and $\mathbf{v}_2 = (r_2, \phi_2)$ are encoded as phasors

$$f_{\mathbf{v}_1} = r_1\ \cos(\omega t + \phi_1)$$
$$f_{\mathbf{v}_2} = r_2 \cos(\omega t + \phi_2)$$

then the encoding of the sum $\mathbf{v} = \mathbf{v}_1 + \mathbf{v}_2 = (r, \phi)$, with $r^2 = r_1^2 + r_2^2 + r_1 r_2 \cos(\phi_1 - \phi_2)$ and $\tan(\phi) = [r_1\ \sin(\phi_1) + r_2 \sin(\phi_2)]\ /[r_1 \cos(\phi_1) + r_2 \cos(\phi_2)]$, is the sum of the phasor encodings of $\mathbf{v}_1$ and $\mathbf{v}_2$:

$$f_{\mathbf{v}} = f_{\mathbf{v}_1} + f_{\mathbf{v}_2} = r \cos(\omega t + \phi)$$

O'Keefe (1991) and Burgess et al. (1992) postulated that such a phasor encoding might be used by rats to encode direction and distance of landmarks. As a reference signal to measure the phase ϕ they proposed the 7–12 Hz EEG θ-rythm in the hippocampus.

Touretzky et al. (1993) and Redish et al. (1993) argued that such a temporal representation of the phase signal has some severe disadvantages. Instead they proposed a spatial representation of the phasor: The continuous, time-dependent signal is distributed over N units. Unit i encodes the amplitude of the sinusoidal wave for the phase $2\pi i/N$. So to encode a vector (r, ϕ) the ith element of the array has an activity $f(i)_{r,\phi} = r \cos(\phi + 2\pi i/N)$. Touretzky et al. (1993) called this architecture a *sinusoidal array*. It is easy to interpret this structure as a neural network. Every unit of the sinusoidal array is realized physiologically as a collection of neurons. Its activity level is represented by the neurons' average firing rate, or the average percentage of neurons firing at any instant. In order to be able to represent the negative half of the sinus wave, the neurons in the sinusoidal array have a fixed baseline firing rate b_0. So the activity of the ith neuron in a neuron group representing the vector (r, ϕ) is

$$f^{\text{neuron}}_{r,\phi}(i) = k_0[r \cos(\phi + 2\pi i/N)] + b_0 \tag{5}$$

where k_0 is a gain parameter.

The advantage of the sinusoidal array encoding of the vectors is that vector addition and subtraction become very simple:

$$f_{\mathbf{v}_1+\mathbf{v}_2}(i) = f_{\mathbf{v}_1}(i) + f_{\mathbf{v}_2}(i) - b_0 \tag{6}$$
$$f_{\mathbf{v}_1-\mathbf{v}_2}(i) = f_{\mathbf{v}_1}(i) + f_{\mathbf{v}_2}(i + N/2 \text{ mod } N) - b_0 \tag{7}$$

for $i = 0,\ \ldots,\ N$.

This means that for vector addition we need only to add the activities of corresponding neurons, i.e. neurons encoding the same part of the phasor (Fig. 2).

Rotation of vectors is possible by means of shifter circuits (Anderson and Van Essen 1987). But this operation is not necessary for path integration. For details see Touretzky et al. (1993) and Redish et al. (1993).

Touretzky et al. (1993) analyzed this architecture with a spiking, noisy neuron model. They found that a stable representation of vectors is possible with this encoding also for noisy neurons. Here we rely on these results and use the above-mentioned average firing rate model.

5 A neural network for path integration

As sinusoidal arrays are well suited for vector computations it seams reasonable to use this encoding for a network model of the path integration process. We investigated how a neural system can use this representation to simplify the computations required to do path integration, although there is no experimental evidence that insects really use this representation.

Before we discuss the details of this model we have to make some assumptions as to how the types of sensory information which are necessary for path integration – compass direction and speed – are represented in the insect's brain.

We suggest that the compass information is encoded by a peak of activity on a dedicated group of neurons. The position of the peak corresponds to the compass direction

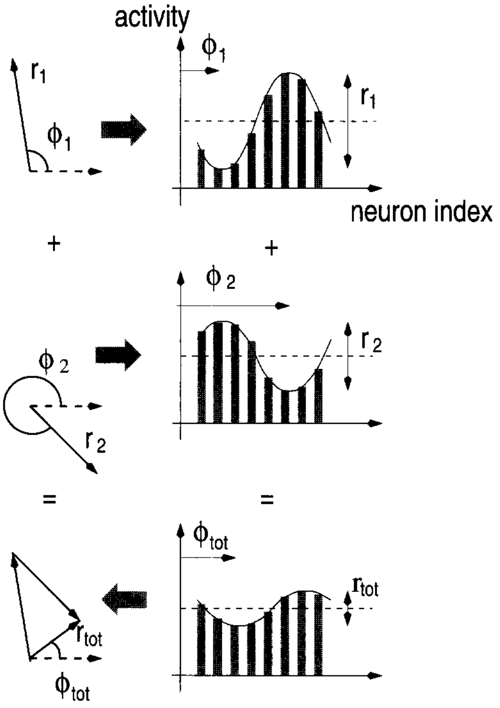

Fig. 2. Addition of vectors in sinusoidal array encoding: If both vectors v_1 and v_2 are encoded as sinusoidal arrays, the vector sum $v = v_1 + v_2$ is computed by adding the activities of the corresponding elements in the arrays

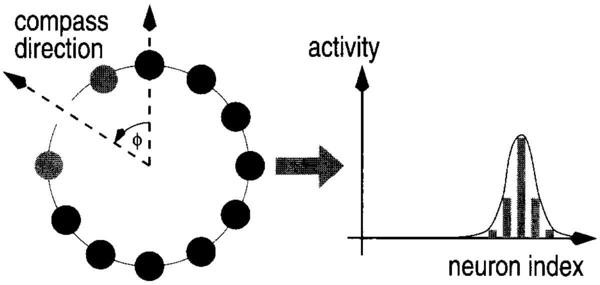

Fig. 3. Representation of compass direction as an activity pattern on a circular group of neurons

(Fig. 3). The neurons are closed in a ring to avoid a discontinuity in the encoding of angles smaller and greater than 2π. Such a representation might easily be computed from the visual input an insect receives. The compass direction is the direction with the least ultraviolet radiation and the most infrared (Wehner 1991). For the representation of speed we assume that it is encoded by the firing rate of a dedicated neuron or neuron group.

There must be some type of short-term memory to store the computed home vector. This information must be kept in neural excitation patterns, because it is not plausible that the rapidly changing representation of the home vector is stored by modification of synaptic strengths.

So we have to investigate:

- how the sensory information can be transformed into a sinusoidal array representation to allow easy computations,
- how the computed home vector can be stored robustly and efficiently,
- how the computed information can be read out to generate the home trajectory.

5.1 Transformation of sensory input

If the compass direction is given by a peak of activity on a dedicated group of neurons, this representation must be transformed into a sinusoidal array representation. Let us assume there is a neuron layer *input*(i) with a distribution of activity which has a maximum, that encodes compass direction ϕ_0 (Fig. 3).

The input pattern has to be transformed into a sinusoidal array with a phase ϕ_0. This can be achieved by connections

$$w_{i,j}^{input} = -k_0 \cos\left[2\pi\left(\frac{i}{N^{input}} - \frac{j}{N}\right)\right] \qquad (8)$$

With these connections the input pattern is transformed into a sinusoidal array which encodes a vector pointing in the direction ϕ_0. To see this let us ignore the minus sign in (8) and remember that for every i, *input*$(i)w_{i\,j}^{input}$ is a sinusoidal pattern encoding a vector with direction $2\pi(i/N^{input})$ and length *input*(i). The sum of all these sinusoidal patterns is a sinusoidal array that encodes the sum of all vectors $\left[2\pi(i/N^{input}), \mathit{input}(i)\right]$. If *input*$(i)$ is symmetrical with respect to ϕ_0 and has a maximum in this direction, this sum encodes a vector pointing in the direction ϕ_0.

The negative sign of $w_{i,j}$ is somewhat arbitrary. It results in calculating the negative of the vector sum of all movements of the animal. This means the network computes the vector from the current location to the starting point, which simplifies the read-out mechanism.

So far we have not considered velocity. Wehner and Srinivasan (1981) showed that ants walk at a constant speed, at least while they are searching for their nest. So maybe the animals' speed need not be considered in a model of path integration. Nevertheless it is possible to include this. To that purpose the vector signal *input*(i) has to be weighted by the speed signal. How can this be done in our neuron model? We can think of multiplying synapses (σ–π units) which modulate $w_{i,j}^{input}$ proportional to v. But there is a simpler method: It is sufficient if the rate with which the compass information is added to the sinusoidal array is modulated with the speed v. In the average firing rate model this both can be described by $k_0 = k_0(v) = v/v_0$.

Finally let us discuss what might happen if the activity distribution of the compass neurons has more than one peak of activity. Experiments which might result in such a situation are discussed in Mittelstaedt (1985). Spiders use the position of a light source as compass information. One might expect that they are disoriented after an outbound excursion in the light of two identical light sources placed in opposite directions. This should be the case if they use this light pattern for reference without further preprocessing. The same would be true for our model if such an activity distribution were present on the compass neurons.

Yet it seems useful to preprocess such an activity pattern with a winner-takes-all network (e.g. Kohonen 1982). Lateral connections which excite the neighboring neurons and inhibit the more distant ones result in formation of a single pronounced peak at the position of the strongest input activity. This results in an input signal which depends only on the direction of the brightest light spot.

If two identical peaks of activity are present, small fluctuations (i.e. chance) decide at which position this peak is formed. In this case the navigation system would produce a vector pointing either in the correct homing direction or in the opposite direction. This is exactly the situation which can be observed in experiments with *Tegenaria* and *Agelena* (Mittelstaedt 1985).

5.2 Short-term memory

In order to use a neural network as a short-term memory we need some kind of feedback connections $w_{i,j}^{mem}$.

$$m_i(t+1) = \sum_{j=1}^{N} w_{i,j}^{mem} m_j(t) = w^{mem} \cdot m(t) \tag{9}$$

If the connections $w_{i,j}^{mem}$ act as a memory, $m_i(t)$ must not change over time. We assume that $w_{i,j}^{mem}$ is symmetric and has eigenvalues $\lambda^{(1)}, \cdots, \lambda^{(N)}$ with $|\lambda^{(1)}| \geq |\lambda^{(2)}| \geq \cdots \geq |\lambda^{(N)}|$ and eigenvectors $a^{(1)}, \cdots, a^{(N)}$. Then an arbitrary pattern $m(0)$ can be written as a linear combination of the eigenvectors $a^{(i)}$: $m(0) = \sum_{i=1}^{N} \alpha^{(i)} a^{(i)}$. So after T iterations of (9) we have

$$\begin{aligned} m(T) &= (w^{mem})^{(T)} \cdot m(0) \\ &= (w^{mem})^{(T-1)} \cdot \left(\sum_{i=1}^{N} \alpha^{(i)} \lambda^{(i)} a^{(i)} \right) \\ &= \sum_{i=1}^{N} a^{(i)} \left(\lambda^{(i)} \right)^{N} a^{(i)} \end{aligned}$$

For large T we get $\lim_{T \to \infty} m(T) = \sum_{i=1}^{K} \alpha^{(i)} a^{(i)}$ if $\lambda^{(1)} = \lambda^{(2)} = \cdots = \lambda^{(K)} = 1$. For $\lambda^{(1)} > 1$ $|m| \to \infty$ and with $\lambda^{(1)} < 1$ $m \to 0$.

So $w_{i,j}^{mem}$ can work as a memory only if its largest eigenvalue is 1. It can store only patterns that are a linear combination of its eigenvectors to the eigenvalue 1. If we choose $w_{i,j}^{mem} = \delta_{i,j}$, all eigenvalues are identical to 1 and we can store arbitrary patterns. For the task of storing vectors encoded as sinusoidal arrays it is optimal to have connections

$$w_{i,j}^{mem} = 2\cos[2\pi(i-j)/N] \tag{10}$$

because in this case $w_{i,j}^{mem}$ has two eigenvectors with eigenvalue 1 and every pattern of (5) can be written as a linear combination of these two eigenvectors. That means that only correct information can be stored in the memory [cf. (9)]; every other pattern will be reduced to the projection onto the subspace spanned by these eigenvectors.

We now have the complete equations for the path integration neurons (Fig. 4):

$$sa(i,t+1) = f\left[\sum_{j=0}^{N-1} \left(w_{i,j}^{mem} sa(j,t) + w_{i,j}^{input} input(j,t) + b_0 \right) \right] \tag{11}$$

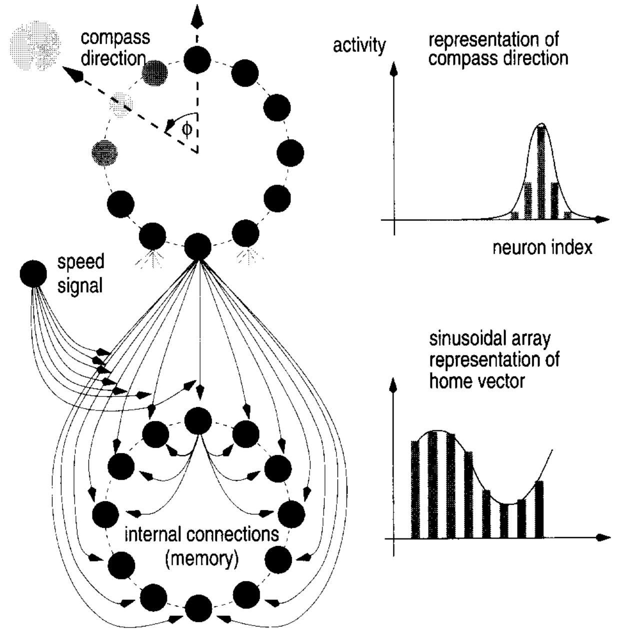
compass direction
ϕ
activity
representation of compass direction
neuron index
speed signal
sinusoidal array representation of home vector
internal connections (memory)

Fig. 4. Summary of the model. The compass direction is encoded as an activity pattern on a dedicated neuron group; this pattern is transformed into a sinusoidal array encoding by connections (8) and added to the vector computed so far. For consideration of speed the connections between both neuron groups have to be modulated by a speed signal

with

$$\begin{aligned} w_{i,j}^{mem} &= 2\cos\left[2\pi\left(\frac{i}{N} - \frac{j}{N}\right)\right] \\ w_{i,j}^{input} &= -k_0(v) cos \left[2\pi\left(\frac{i}{N^{input}} - \frac{j}{N}\right)\right] \\ k_0(v) &= \frac{v}{v_0} \end{aligned} \tag{12}$$

$$f(x) = \begin{cases} 0 & \text{if } x < 0 \\ x & \text{if } 0 < x < 2b_0 \\ 2b_0 & \text{if } 2b_0 < x \end{cases} \tag{13}$$

Because of the sigmoidal function f this system works only for distances smaller than a fixed distance R_0 for which the argument of f becomes greater than $2b_0$ and part of the sinusoidal pattern will be disrupted. But this does not represent a serious limitation as the exploration range of insects is restricted in any case.

5.3 Read-out of computed home vector

For the animal to get back to its home it needs only to know in which direction to turn to go straight back to its nest. If the path integration system keeps on integrating, the animal knows it has reached its nest if its home vector is zero.

To generate a turn signal, the compass information and the home vector have to be compared. With connections like those in (8) the animal moves straight home if the peaks of activity in both neuron arrays are at corresponding positions.

So we need neurons that are sensitive to a displacement of these two peaks to the left or to the right. These neurons must have receptive fields on both neuron arrays that are

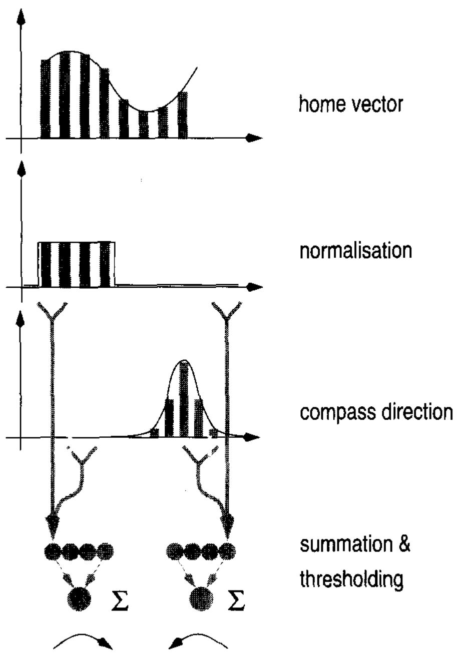

Fig. 5. Read-out mechanism. A turn signal that turns the animal into the homewards direction can be derived by comparison of the sinusoidal array representation of the home vector and the encoding of the compass direction; if the maxima of both patterns are at corresponding positions the animal is oriented homeward. A turn signal is generated by neurons that are sensitive to a certain displacement of the maximum on one neuron group with respect to the other

displaced relative to each other. If their threshold is high enough so that they can be activated only if they receive input from both arrays, they are sensitive to a certain displacement of both patterns of activity and can generate the appropriate motor signal.

To ease the comparison of the two peaks the sinusoidal array is normalized first in an intermediate layer (Fig. 5).

6 Experimentally observed deviations from the direct homing course

The model described in the previous section shows that exact path integration is possible with a neural network and is easy to achieve. Nevertheless Müller and Wehner (1988) found significant deviations of the ants' homing direction from the direct course back to the nest. Similar deviations were found in experiments with other insects (Wehner 1991) and also in experiments with small mammals (Seguinot et al. 1993).

Müller and Wehner (1988) and Hartmann and Wehner (1995) interpreted these errors as a clue to computational limitations in the ant's brain and derived models of the integration process that showed the same errors. But if we analyze these models we find that in both models the free parameters are not tuned to achieve the optimal performance of the described architecture. For instance in the model of Müller and Wehner (1988), (1) and (3) were approximated by:[1]

$$l_{n+1} = l_n + l_0 \left(1 - \frac{|\delta_n|}{\pi/2}\right) \tag{14}$$

$$\phi_{n+1} = \phi_n + k\,(\pi + \delta_n)(\pi - \delta_n)\delta_n \frac{l_0}{l_n} \tag{15}$$

with unit length $l_0 = v_n \Delta t_n$. The free parameter k was chosen to fit the experimental data. This fit resulted in $k_{\text{exp}} = 0.1316$. If k was chosen so that $k(\pi + \delta)(\pi - \delta)\delta$ provides an optimal approximation to the sine function we would get $\tilde{k} = 315/4\pi^6 \approx 0.0819$ (by minimizing the mean square error in the interval $[0, 2\pi]$). So the parameter derived from the experimental data differs significantly from the optimal value $\tilde{k}$ that would result in the most accurate computation possible by this model. Using $\tilde{k}$ instead of k_{exp} would result in maximum errors of about 5°.

The same argument holds for the model of Hartmann and Wehner (1995). The free parameters of this model are also chosen to fit the experimental data and not to achieve optimal performance from the proposed architecture.

We conclude that it is unlikely that the deviations observed by Müller and Wehner (1988) can be explained by systematic errors in the path integration system of the ant. Instead there must be some advantage for the ant to choose a homing course different from the straight path to the nest.

6.1 Simplification of landmark navigation by choosing an indirect return path

Choosing a return path different from the direct course to the nest might be an advantage for the animal if we consider that ants use landmark navigation to pinpoint the nest after the approximate position of the nest has been reached (Wehner 1991). For landmark navigation it is advantageous to approach the goal location from a fixed direction. Then known landmarks appear always on the same position of the retina. Indeed Collett and Baron (1994) found that bees prefer special viewing directions, when they are approaching a previously learned position. And Zeil (1993a,b) found similarities in the orientation flight and the following return flights of wasps, indicating that these animals try to see the same sequences of landmarks during a return flight as they have learned during their orientation flight.

If an animal tries to approach its nest preferably from one special direction it should not aim directly towards its nest but to a point near the nest from were it can always approach the nest in approximately the same direction. If it were to walk straight to the nest it might end up in any direction relative to the nest. If it first approaches a point in the vicinity of the nest it can be sure to reach the nest from a fixed direction (Fig. 6).

This effect can be obtained if the ant begins its integration process not directly at the nest but after travelling for a certain distance k. Figure 7 shows a simulation of this assumption for the two-leg experiments from Müller and Wehner (1988). The distance k that the ant travels before it starts its integration process can be adapted to fit all three experimental situations described in Hartmann and Wehner (1995).

This assumption is also consistent with the observation of Müller (1989) that the return trajectories observed in his

[1] Equations (14) and (15) differ slightly from the equations in Müller and Wehner (1988): we use radians instead of degrees, and a unit length l_0 instead of $l_0 = 1$.

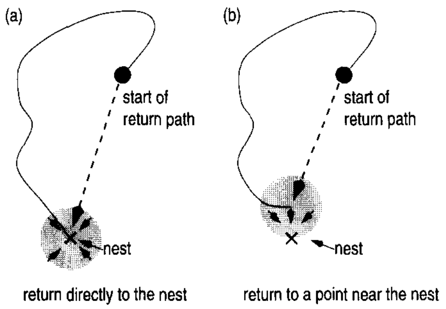

Fig. 6. Comparison of two homing strategies: **a** Start path integration at the nest (cross) **b** Start path integration at a point near the nest. If an ant returns directly to the nest (**a**) it must expect to end up at any place somewhere around the nest (gray shaded area). Thus it has to approach the nest from different directions every time. If it aims towards a certain point in the vicinity of the nest (**b**), it also ends up only in the neighborhood of this point, but nevertheless it always approaches the nest from approximately the same direction

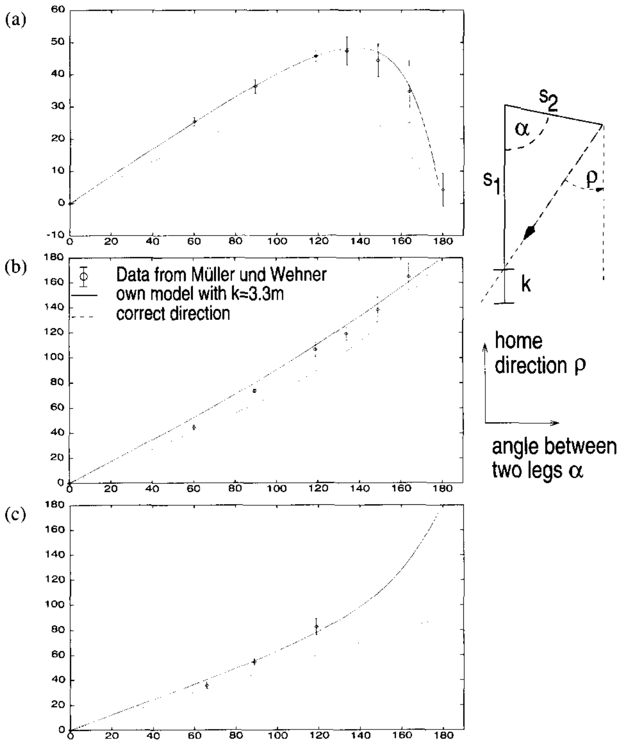

Fig. 7. Comparison of experimental data from Hartmann and Wehner (1995) with the hypothesis that ants start their integration process not at the nest but after traveling a certain distance k. **a** s_1 = 10 m, s_2 = 5 m; **b** s_1 = 5 m, s_2 = 10 m; **c** s_1 = 10 m, s_2 = 10 m

experiments with ants were systematically shorter than the distance from the release site to the nest.

A similar assumption was formulated by Seguinot et al. (1993) who investigated path integration in golden hamsters. Their experiments were in good correspondence to the model of Müller and Wehner (1988) except for the case when the animals were forced to perform an outbound trajectory which contained a loop that brought the animal back to the former nest location and then to a position away from the nest. This indicates that the deviation of the observed homing course from the direct path back to the nest depends on the position of the animal relative to the nest and not on the animal's path to this point, as suggested by the model of Müller and Wehner (1988).

7 Discussion

We have presented a model which describes how path integration might be implemented in an insect's brain. It is plausible that this is the mechanism used by insects because it is easy to construct in the framework of generally accepted neuron models. According to this model a neural network can easily compute a home vector without the need for approximations.

This is contrary to the model of Müller and Wehner (1988), who proposed that ants approximate the trigonometric functions with polynomials, and to the model of Hartmann and Wehner (1995), in which these functions were approximated by polygons. In our model the computation is simplified by choosing an appropriate representation. With this representation it becomes easier to do the exact computation than an approximation with polynomials or polygons.

As Müller and Wehner (1988) found that ants make substantial errors during their path integration process, the following question remains: If path integration with neurons is easy to accomplish without the need of approximations why do insects make the observed errors in computing their homeward trajectory? We have been able to show that these apparent errors can be explained by a simple strategy to simplify landmark navigation after the approximate position of the nest is reached.

How does this approach to path integration compare with the models by Mittelstaedt (1985, 1989)? Similar to Mittelstaedt's approach we conclude that path integration is performed without approximations. In his bi-component model he proposes that signals proportional to the sine and to the cosine of the animal's orientation with respect to a fixed reference direction are generated. These signals are summed to compute the x and y coordinates of the animal's location relative to the starting point.

If we reduce the number of neurons in the sinusoidal array in (5) to four, our model becomes similar to Mittelstaedt's bi-component model, since these neurons then compute $\sum \cos(\phi_n - 2\pi k/N)$ for $k = 0, \ldots, 3$, i.e. $\sum \sin \phi_n$, $\sum \cos \phi_n$, $-\sum \sin \phi_n$, and $-\sum \cos \phi_n$. The synaptic weights which are used to compute these values are similar to the connectivity proposed in the map-weighting model of Mittelstaedt and Eggert (1989).

If more neurons are used, more components $\sum \cos(\phi_n - \Delta)$ are calculated. Thus the components of the home vector are determined not only for the minimum number of two orthogonal directions. Instead a larger number of components is used, which results in a more robust and redundant coding. Another advantage of such a representation is that it is easier to read out the computed home vector. In order to generate the turn signal necessary to orient the animal to its nest, only the positions of the activity peaks of two neuron groups have to be compared. This is easy to accomplish according to standard neuron models.

References

Anderson C, Van Essen D (1987) Shifter circuits: a computational strategy for dynamic aspects of visual processing. Proc Nat Acad Sci USA 84: 6297–6301

Burgess N, O'Keefe J, Recce M (1992) Using hippocampal 'place cells' for navigation, exploiting phase coding. From Neuroprose Archive; can be retrieved by anonymous ftp from ftp://archive.cis.ohio-state.edu/pub/neuroprose

Collett T, Baron J (1994) Biological compasses and the coordinate frame of landmark memories in honeybees. Nature 368: 137–140

Görner P, Claas B (1985) Homing behaviour and orientation in the funnel-web spider *Agelena labyrinthica.* In: Barth F (ed) Neurobiology of arachnids. Springer, Berlin Heidelberg New York, pp. 277–297

Hartmann G, Wehner R (1995) The ant's path integration system: a neural model. Biol Cybern, this issue

Kohonen T (1982) Self-organized formation of topologically correct feature maps. Biol Cybern 43: 59–69

Mittelstaedt H (1985) Analytical cybernetics of spider navigation. In: Barth F (ed) Neurobiology of arachnids. Springer, Berlin Heidelberg New York, pp. 298–315

Mittelstaedt H, Eggert T (1989) How to transform topographically ordered spatial information into motor commands. In: Arbib M, Ewert J-P (eds) Visuomotor coordination: amphibians, comparisons, models, and robots. Plenum Press, New York, pp. 569–585

Mittelstaedt H, Mittelstaedt M-L (1982) Homing by path integration. In: Papi F, Wallraff HG (eds) Avian navigation. Springer, Berlin Heidelberg New York

Mittelstaedt M-L, Glasauer S (1991) Idiothetic navigation in gerbils and humans. Zool Jahrb Physiol 95: 427–435

Moller P, Görner P (1994) Homing by path integration in the spider *Agelena labyrinthica* Clerck. J Comp Physiol [A] 174: 221–229

Müller M (1989) Mechanismen der Wegintegration bei *Cataglyphis fortis* (Hymenoptera, Insecta). PhD thesis, University of Zürich

Müller M, Wehner R (1988) Path integration in desert ants, *Cataglyphis fortis.* Proc Natl Acad Sci USA 85: 5287–5290

O'Keefe J (1991) An allocentric spatial model for the hippocampal cognitive map. Hippocampus 1: 230–235

Redish A, Touretzky D, Wan H (1993) The sinusoidal array: a theory of representation for spatial vectors. In: Proceedings of the Computational Neurosciences Conference CNS 93. Kluwer Academic Publishers, Boston

Rumelhart D, McClelland J (1986) Parallel distributed processing, vols. 1 and 2. MIT Press, Cambridge, Mass

Schäfer M, Wehner R (1993) Loading does not affect measurement of walking distance in desert ants, *Cataglyphis fortis.* Verh Dtsch Zool Ges 86: 270

Seguinot V, Maurer R, Etienne A (1993) Dead reckoning in a small mammal: the evaluation of distance. J Comp Physiol [A] 173: 103–113

Touretzky D, Redish D, Wan H (1993) Neural representation of space using sinusoidal arrays. Neural Comput 5: 869–884

Wehner R, (1991) Visuelle Navigation: Kleinstgehirn–Strategien. Verh Dtsch Zool Ges 84: 89–104

Wehner R (1994) The polarization-vision project: championing organismic biology. Fortschr Zool 39: 103–143

Wehner R, Müller M (1993) How do ants acquire their celestial ephemeris function? Naturwissenschaften 80: 331–333

Wehner R, Srinivasan M (1981) Searching behaviour of desert ants genus *Cataglyphis* (Formicidae, Hymenoptera). J Comp Physiol [A] 142: 315–338

Zeil J (1993a) Orientation flights of solitary wasps. I. Description of flights. J Comp Physiol [A] 172: 189–205

Zeil J (1993b) Orientation flights of solitary wasps. II. Similarities between orientation and return flights and the use of motion parallax. J Comp Physiol [A] 172: 207–222