PHILOSOPHICAL TRANSACTIONS OF THE ROYAL SOCIETY A
MATHEMATICAL, PHYSICAL & ENGINEERING SCIENCES

The Mind of a Robot [and Discussion]

Michael Brady, Huosheng Hu and B. Webber

Phil. Trans. R. Soc. Lond. A 1994 **349**, 15-28
doi: 10.1098/rsta.1994.0110

The mind of a robot

By Michael Brady and Huosheng Hu
Robotics Research Group, Department of Engineering Science, University of Oxford, Oxford, U.K.

Work in robotics offers to the study of intelligence insights gleaned from building autonomous agents that operate purposefully in the real world. We sketch the design of such an agent and discuss some of the issues that the design of such agents raises for intelligence and the mind.

1. Introduction

AI has been dominated by cognitive aspects of intelligence and mind: programs deal with symbolic representations of information, prove theorems, diagnose illnesses, play games, attempt to solve practically useful problems such as assessing creditworthiness, and engage in various other forms of intellectual challenge. This has established the themes of AI: the representation and mobilization of knowledge in symbolic structures, reasoning (non-monotonic or otherwise), matching for analogical reasoning, and planning. Common to all these endeavours is that the computer has a restricted interface to the real world, typically a user that types questions and inputs knowledge.

There has been an exception, though it has always remained on the fringes of AI: sensory processing, particularly vision and speech understanding, and robotics. Here the system derives its essential usefulness and power from continual interaction with the real external world. From the very start of the enterprise of building such systems, it was realized that there is no getting around the fact that the world is a complex continually changing place (save in psychological laboratories), and signals are inherently noisy so that nothing is certain. The methods and methodologies of robotics (in which we include sensory processing) diverged from those of cognitive AI. Uncertainty was accommodated using probabilities and bayesian methods, and continuous mathematics was to the fore, for example in extracting intensity changes, or in filtering auditory signals.

Early AI work in robotics and vision aimed to get this tiresome business out of the way as soon as possible, to get on with the 'real' work of constructing and matching structured symbolic representations of scenes and prior models. This strict separation into a kind of engineering front-end processing followed by more 'AI-like' reasoning has not proved to be reliable, though it has enjoyed some success. Nine years ago, in a keynote address to the American Association for Artificial Intelligence (Brady 1985), one of us argued that there must be a closer synergy between work in AI and in robotics. This paper will continue to develop that theme. First, to be intelligent, robotics must embrace developments in Artificial Intelligence. Brady characterized robotics as the 'intelligent connec-

Phil. Trans. R. Soc. Lond. A (1994) **349**, 15–28
Printed in Great Britain

TEX Paper

tion of perception to action'. By this, I meant that AI processing was required to mediate between raw signals and the generation of motor commands. This did not imply, of course, that all sensory processing should be complete before AI processing and, in turn, before the generation of motor commands. Conversely, robotics provides a natural and challenging testbed for AI, because it forces systems to interact with the real world. Work of that sort has, in fact, been the most innovative contribution to AI over the past decade, and we return to it below.

Let us begin by stating what may seem obvious, though it equally may seem perverse that we state it at a meeting concerned with 'the mind': the primary requirement of any living system is to survive, and, through procreation, to ensure the survival of its species. The key challenges faced by a living system are to find food, shelter and a mate, and to recognize and respond to threats and opportunities offered by the environment. Clearly, its responses should be timely: a system that becomes rooted to the spot to contemplate the significance or otherwise of an entity that may pose a threat to its well-being is unlikely to survive long or to have evolutionary staying power.

That this remark may be considered perverse at the present meeting is because the mind is normally equated with loftier intellectual pursuits: proving mathematical theorems, studying philosophy, or doing the *Times* crossword. A common view is that the abilities referred to in the previous paragraph are 'simple reflexes', below the level of attainment that we might consider intelligent. However, a remarkable percentage of the brain, as measured by weight, is dedicated to sensory processing and motor control. One estimate puts the figure at 93%. Even if that figure is only roughly correct, it is clear that humans are primarily seeing–moving beings, constantly interacting with an ever-changing environment. According to this view, human intelligence evolved to offer competitive advantage primarily at tasks of the sort outlined in the previous paragraph. The ability to solve IQ puzzles, do crosswords and write poetry is a relatively recent luxury. Although we may feel good about our ability to do such things, it is not what our brains were developed for. In short, robotics is not peripheral to the study of intelligence; it is central.

The construction of artificial systems has the potential to provide fresh insights about intelligence. It need not do so, of course, because a narrowly applications-focused system may achieve performance but be based on a design (sometimes called an architecture) that does not generalize well and whose performance stems primarily from technology, for example a vast increase in computing power. If it is not sufficient to build such systems, is it at least insightful to do so? We hope to convince the reader that the answer is yes.

Before proceeding to our themes, let us note that AI, in common with other emerging sciences, is prone to fashion. Proponents of a new approach often overstate the import of their contributions, not least by denouncing all previous work. We should be clear from the outset that the baby that is work in robotics need not displace much of the bathwater of AI. For example, the ability of a robot to respond reactively to threat does not imply that there is no need for planning. Advances in the mathematics of early vision do not imply that there is no need for stored representations of shapes, nor that those representations cannot be composed of symbolic structures.

The remainder of this paper examines a number of themes that have been explored in robotics and which we argue are important aspects of any theory of mind: (i) reactivity and planning; (ii) the requirement for distributed processes; (iii) the need to cater for uncertainty; (iv) purposive behaviour and (v) emergent properties from interacting processes.

2. Planning and reactivity

Planning is central to purposive behaviour. To plan is to formulate a strategy to achieve a desired state of affairs – in the jargon, to attain a goal. An inability to plan implies purely reflexive responses, with an agent stumbling aimlessly about in response to unexpected external events. However, planning has traditionally been treated as a purely predictive process where complete knowledge is assumed *a priori* about the environment either in the form of a qualitative or a quantitative model. A complete sequence of steps that solves a problem or reaches a goal is determined before taking any action. Recovering from error and dealing with unexpected events is usually left as a subsequent execution stage. As a result, traditional planning systems are inherently open-loop and unable to handle unmodelled disturbances in the real world. Recently, research on planning has focused on the ability of a planner to cope with a dynamic environment, under the name of reactive planning. This is because unforeseen changes in the world, coupled with uncertainty and imperfect sensory information, force an agent (or a robot system) to plan and react effectively under time constraints. In other words, planning must evolve over time, through cycles of sensing, planning and action, updating the internal model dynamically when new information becomes available.

Planning embraces a broad range of abilities, and for our present purposes it is useful to distinguish four: *mission planning* is the process of determining the requirements and constraints for the global tasks and obtaining an optimal schedule for multiple goals. *Global path planning* aims to find a collision-free path (a set of intermediate points) for a mobile robot to follow from the start position to the goal position. In contrast, *local path planning* generates relatively local detours around sensed obstacles while following a globally planned path. Finally, *trajectory planning* generates a nominal motion plan consisting of a geometrical path and a velocity profile along it in terms of the kinematics of the individual robot.

What we call mission planning is what is normally called planning in cognitive AI. For example, planning a journey to Koh Samui, subject to constraints (e.g. to get there within a week), probably involves flying via Bangkok rather than walking. Planning the assembly of a complex device such as a car engine inevitably involves hierarchical decomposition into substructures, each planned separately but respecting interfaces. Military mission planning involves the disposition of forces, continually monitoring enemy movements, and anticipating and then countering threats.

Global path planning for a mobile robot in a known environment with known static objects has been studied extensively. Graph searching and potential field methods are the two main approaches used to solve the path-finding problem. The main feature of both methods is that the entire environment is modelled geometrically before the motion takes place. In the graph searching approach, a graph is created that shows the free spaces and forbidden spaces in the environment of the robot. A path is then generated by piecing together the free spaces or by tracing around the forbidden area. In contrast, the potential field approach uses a scalar function to describe both objects and free space. The negative gradient of the potential field gives precisely the direction to move to avoid obstacles. It offers a relatively fast and effective way to solve for safe paths around obstacles.

However, the environment of a robot is not always static. Dynamic changes may be due to the motion of the robot, the appearance and disappearance of objects, and to object motion. If the changes are predictable, they can be taken into account when the robot plans its optimal path. But if the world changes unpredictably, the robot has to plan and replan a collision-free path dynamically. This is the problem of planning under uncertainty. In such a case, a robot has to rely on its sensors to detect unexpected events and then adapt its path accordingly. Further, to plan an optimal path, uncertain events, such as unexpected obstacles, should be quantified as costs in order to make judgement feasible. We return to this below.

In real life, in both mission planning and global path planning, we need to react to unforeseen events, modify our plans accordingly, contending all the while with uncertain information, and do so against constraints such as time. In every instance of planning in the real world, the best laid plans do indeed 'gang aft a-gley' (often go wrong).

The earliest influential planning system, STRIPS (Fikes & Nilsson 1972), planned a sequence of moves through a set of rooms to reach a desired location. Unlike most subsequent planners, STRIPS was connected to a plan execution software module, PLANEX, which orchestrated the execution of the planned path devised by STRIPS on a sensing mobile robot, SHAKEY. SHAKEY's sensors could detect unexpected obstacles, causing PLANEX to suspend its script, halting the robot, so that STRIPS could be invoked afresh in the new situation. In this way, SHAKEY's behaviour alternated mindless script following and long periods of planning, which was regarded as a form of reasoning.

Recent work in planning has aimed to overcome this limitation. A variety of proposals have been made to develop the idea of a situated agent (Agre & Rosenschein 1994), a software object intended to be in continuous interaction with the world. Sadly, the vast majority of situated agents only inhabit a simulated world in which the problems of noise, uncertainty and clutter are absent (but see Rosenschein & Kaelbling (1986) for a notable exception). We return to agents, and to the software architectures developed for them, in the next section.

Our work has concentrated on path planning for a mobile robot. The first issue being addressed is how to model the dynamic, uncertain environment in a manner that makes it possible to provide a solution for an optimal path constrained by the real world. Most approaches use active on-board sensors for the acquisition of information about the robot's surroundings, of which various grid representations have been proposed. In such approaches, probabilistic models are built based on grid cells and updated dynamically using sensor data. A path is then found by minimizing the probability of encountering obstacles. However, such methods require enormous computation to set up different grids to map the environment, forcing the robot to deal with huge amounts of data. Moreover, if the kinematics and dynamics of a non-holonomic mobile robot are taken into account, the method is difficult to implement. Because there is not complete information about the robot's environment during planning, the methods achieve a suboptimal solution in most cases.

PHILOSOPHICAL TRANSACTIONS OF THE ROYAL SOCIETY A MATHEMATICAL, PHYSICAL & ENGINEERING SCIENCES

We have proposed (Hu & Brady 1994*b*) a probabilistic approach to address the problem of path planning with uncertainty for mobile robots. Instead of using an uncertainty grid, we use a topological graph to represent the free space of the robot's environment, weighted by scalar cost functions. The statistical models are built to quantify uncertainty, forming uncertain costs for unexpected events. These uncertain costs are updated by available sensor data when the robot moves around. An optimal path is then found from the topological graph using cost functions and dynamic programming.

Analogous to the comment about mission planning made earlier, in traditional planning systems, global path planning and local path planning are sequential processes. In other words, the local planner is idle when the global planner is busy, and vice versa. This causes a delay for the whole system, because whenever a preplanned path is blocked, the local planner triggers the global planner and then has to wait for an alternative path. Concurrent processing of both planners is crucial for avoiding delays. In our design, however, the global planner generates alternative subgoals dynamically when the robot is travelling along a preplanned optimal path. In other words, it is always assumed that the next node of the preplanned path may be blocked by unexpected obstacles. If, however, this turns out to be true, the local planner can backtrack along these subgoals without delay. However, if nothing happens when the robot is approaching the next node, the alternative subgoals provided by the global planner will be ignored. The system has been implemented and performs real-time local and global path planning and obstacle avoidance. Dynamic replanning is performed as necessary, based on decisions that are rooted in sensory information.

3. Distributed processing

Any robot system or autonomous mobile robot needs constantly to process large amounts of sensory data in order to build a representation of its environment and to determine meaningful actions. The extent to which a control architecture can support this enormous processing task in a timely manner is affected significantly by the organization of information pathways within the architecture. The flow of information from sensing to action should be maximized to provide minimal delay in responding to the dynamically changing environment. A distributed processing architecture offers a number of advantages for coping with the significant design and implementation complexity inherent in sophisticated robot systems. First, it is often cheaper and more resilient than alternative uniprocessor designs. More significantly for this meeting, multiple, possibly redundant, processors offer the opportunity to take advantage of parallelism for improved throughput and for fault tolerance. Note that we distinguish the design of a processing structure (architecture) from its realization in hardware and/or software.

Over the past two decades, a good deal of thought and effort has been dedicated to the design of architectures to tame complexity and achieve new heights of performance. Two principal designs have been adopted: the functional and behavioural decomposition (Brooks 1989).

Functional decomposition follows the classic top-down approach to building systems. The entire control task of a mobile robot is divided into subtasks which are then implemented by separate modules. These functional modules form a

PHILOSOPHICAL TRANSACTIONS OF THE ROYAL SOCIETY A MATHEMATICAL, PHYSICAL & ENGINEERING SCIENCES

chain through which information flows from the robot's environment, via the sensors, through the robot and back to the environment via actuators, closing the feedback loop. Most previous mobile robots have been based on this approach, including, for example, hierarchical and blackboard architectures; but both of these have inherent limitations, including poor use of sensory information, reduced bandwidth causing bottlenecks, and difficulty in dealing with uncertainty.

In contrast, behavioural decomposition is a bottom-up approach to building a system. A behaviour encapsulates the perception, exploration, avoidance, planning and task execution capabilities necessary to achieve one specific aspect of robot control. That is, each is capable of producing meaningful action, and several such can be combined to form increasing levels of competence (Brooks 1989). In other words, each of them realizes an individual connection between some kind of sensor data and actuation. The system is built step by step from a very low level, say from locomotion, to obstacle avoidance, to wandering. Successive levels can be added incrementally to enhance the functionality of the robot.

This design method has grown in popularity recently (Hu & Brady 1994*a* provide references). In the subsumption architecture, for example, control is distributed among those task-achieving behaviours that operate asynchronously. Lower layers can subsume the operation of the higher ones when necessary, only one layer actually controlling the robot at any one time. Because each layer achieves a limited task, it requires only that information which is useful for its operation. It is claimed that the control system can respond rapidly to dynamic changes in the environment without the delays imposed by sensor fusion. But the implementation of higher layers of competence still poses a problem. More careful initial design in specifying the communications and modularity is required. Moreover, the higher levels often rely on the internal structure of lower levels, thus sacrificing modularity.

Neither of these approaches suffices because the control of a mobile robot is so complex that one cannot strictly adhere to one decomposition scheme while completely ignoring the other. Each has benefits and drawbacks. We have developed and implemented (Hu & Brady 1994*a*) an architecture that is, we contend, a blend of the best features of each. It consists of a distributed community of sensing, action and reasoning nodes. Each of them has sufficient expertise to achieve a specific subtask, following a hierarchical decomposition scheme. Few of them are composed of a single task-achieving behaviour as in a behavioural decomposition scheme. The key point is that the control task of the mobile robot is distributed among a set of behaviour experts that tightly couple sensing and action, but which are also loosely coupled to each other. In this way, sensor data can be used directly in a corresponding layered control task to form a task-achieving behaviour. This differs from the functional decomposition in which the combination of subtasks is a single processing chain. To function effectively, each layer requires a significant amount of self-knowledge to allow it to decide what information it can supply to others, how best to recover from local errors, and also what to do if no information is sent from other layers.

Basically, there are two subsystems in our design: a layered perception system and a layered control system. The layered perception system is used for active sensor control and distributed sensor fusion to support a layered control strategy. The layers process raw sensor data from internal sensors (tachometer, encoder, resolver) and external sensors (sonar, vision) to build up models in a bottom-up

PHILOSOPHICAL TRANSACTIONS OF THE ROYAL SOCIETY A MATHEMATICAL, PHYSICAL & ENGINEERING SCIENCES

manner. All the sensing layers operate independently and are loosely coupled. Communication between them is used to realize sensor data fusion. The design of the layered controller is based primarily on the observation that different response times are demanded by different tasks. The lower levels perform simple, general tasks such as smooth path guidance and avoiding obstacles for fast reactivity. The higher levels perform more complex, situation-specific tasks such as path planning and monitoring. All layers operate in parallel.

Each layer of our distributed real-time architecture consists of a control node and a sensing node. Each sensing node delivers a representation, which, in the context of a given task, causes a corresponding control node to generate commands. Here, we try to avoid a centralized and complicated model for a dynamic environment because it needs more time to compute, thereby reducing the system's response speed. Instead, we are using several simple models, each tuned to a different range and resolution of situations for different tasks. In other words, this multi-model control architecture takes the real-time capability and the expected task-achieving behaviours into account.

Issues concerning the realization in hardware or software of distributed architectures such as those described above are almost tangential to the concerns of this meeting, but it may be helpful to point out some links to other contributions. A key issue in realization is granularity: a fine-grained architecture is generally considered to comprise many thousands of individual processing elements, whereas a coarse-grained architecture comprises just a few tens or hundreds. In every case, there is a trade-off between computation and communication. Granularity greatly influences software design. We have adopted a coarse granularity because there is a well developed theory of inter-process communication (Hoare 1985) and commercially available processors to implement it (transputers). Our design is based on a module called LICA (locally intelligent control agent) (see Hu & Brady 1994*a*).

The connection machine offers a currently expensive realization of fine granularity. Neural networks offer an alternative but we have not explored their use much to date, partly because their mathematical foundation is still very much in development, and partly because they are almost without exception realized only in software.

4. Uncertainty

Any robot or animate system that operates continuously in the real world must rely on information provided by its sensors. It is not possible, regardless of how many sensors are used or how discriminating they are, to observe directly everything of interest, all the time, and with complete accuracy: the sensors available may not be able to make direct observations on everything of interest. As we noted in § 2, there is always a limit to the temporal sampling; and all sensors are subject to noise and error. Therefore, a planning system is not able to maintain a complete picture of the environment with complete certainty.

A system which plans under uncertainty must maintain multiple possibilities for the state of the world, and associate with each some degree of belief. Some alternative world states will be more likely than others; for example, the system must allow for the possibility that the sensor data is incorrect. New sensor data about the world must be used to update these possibilities and beliefs; some alternatives may be ruled out and new ones generated, whereas others may become more or less likely.

Let us suppose, for example, that a mobile robot is commanded to travel along a path generated by a global path planner, say the one we described in § 2. While traversing the path, an unexpected obstacle appears and the possibility of a collision arises. Because sensors inevitably have limitations on their range and accuracy, they may not be able to tell exactly whether the gap between the obstacle and a known object is wide enough for a sidestep manoeuvre when the mobile robot is some distance away. A decision is required as to whether the robot should continue its motion along the path to make further observations to manoeuvre (at a certain cost), or, alternatively, to follow a different path and incur a certain loss. If the cost of extra sensing is less than the cost of taking the alternative path, it may be worth persevering along the original path. But the robot may eventually find the gap impassable, incurring an overall cost greater than immediately abandoning the planned path and following the alternative. Hu & Brady (1994*b*) adopt a bayesian decision theoretic approach to this problem. First, a probabilistic model is formulated of the (sonar) sensory information available to the robot. A loss function is defined that provides the outcome of an action (e.g. sidestep) given the path state (e.g. passable). Then that action is chosen which minimizes the Bayes risk.

The bayesian framework is but one of a number of approaches to uncertainty that has been explored in AI. Pearl (1988) makes the following classification of AI approaches to uncertainty: logicist, neo-calculist and neo-probabilist. Logicist approaches use non-numerical techniques for dealing with uncertainty, mainly non-monotonic logics. Neo-calculist approaches use a numerical representation of uncertainty, but invent new calculi, considering the traditional probability calculus inadequate; examples are Dempster–Shafer calculus, fuzzy logic, certainty factors (see Nicholson 1992 for references). The neo-probabilist school, which includes our work, remains within the traditional bayesian framework of probability theory but adds the computational facilities required by AI tasks.

There have been many objections by the AI community to the use of probability, including the observation that people seem to be bad probability estimators. When the planning/navigating system asserts that 'the chances that an object in region X at time T will move to region Y is p', the important thing is not the precise magnitude of p, so much as the specific reason for this belief (the sensor data, its schedule or its previous movement), the context or assumptions under which the belief should be firmly held, and the information which would lead to this belief being revised.

Belief networks allow the representation of causal relations and provide a mechanism for integrating logical inference with bayesian probability. Belief networks are directed acyclic graphs, where nodes correspond to random variables, say world states or sensor observations. The relationship between any set of state variables can be specified by a joint probability distribution. Evidence can be provided about the state of any of the nodes in the network. This evidence is propagated through the network using a bidirectional (message passing) mechanism, affecting the overall joint distribution.

We return to belief networks momentarily, but first we pause to consider an extremely useful first cousin, the Kalman filter. The Kalman filter maintains an estimate of the state x of a system and a covariance estimate P of its uncertainty. Bar Shalom & Fortmann (1988) provide a lucid introduction to the theory of Kalman filtering. The application of the Kalman filter to sensor-guided control is used to reduce the discrepancy between the planned and actual states of a robot vehicle which increase, as does the state uncertainty, when no sensor measurements are made. Suppose, however, that the vehicle senses a planar surface. Consistent with our intuitions, the uncertainty orthogonal to the wall decreases sharply; but continues to increase in the direction tangential to the wall. If a second wall is sensed, the state uncertainty is sharply reduced.

The Kalman filter assumes that the dynamics of the system can be linearized, so that the transformation from the state at the kth time step to the $(k+1)$th are given by a matrix (linear) operation on the state, but corrupted by noise. In its simplest form, the Kalman filter algorithm proceeds as follows (see, for example, Bar Shalom & Fortmann 1988, p. 61): (i) the state and uncertainty are predicted at time $k+1$ on the basis of the system dynamics and previous states; (ii) measurements of the state are then taken. It is expected that these will be corrupted by noise. (iii) Finally, the estimate of the state and uncertainty at time $k+1$ is formed by a linear combination of the prediction and measurement, the exact combination being controlled by the uncertainty, as a measure of the extent to which the predicted state and measured state are to be believed. The Kalman filter has the property that, under certain reasonable assumptions, it is the optimal state estimator. Note that it is not without problems in practice. Among the more severe of these are the difficulties of computing a good *initial* state estimate, of determining appropriate gain matrices and of identifying and approximating real plants in the simple form shown. The Kalman filter has been much used at Oxford (and elsewhere) to guide robot vehicles, track moving shapes, and in computer egomotion.

The Kalman filter is related to the distributed processing discussed in the previous section. In a typical fielded system, a set of sensors make independent measurements of components of the state and report them, at each step, to a central processor that runs the Kalman filter. If one of the sensors ceases to operate, the system continues to run, albeit it with increased state uncertainty. If, however, the central processor ceases to operate, the whole system fails. An obvious alternative design is to distribute the Kalman filter among the sensors (in the current parlance this makes them 'smart' sensors) and enable them to communicate their state estimates amongst themselves. Rao *et al.* (1991) showed that the equations of the Kalman filter can be partitioned so that such a fully decentralized system converges to the same global optimum as the centralized system. The system degrades gracefully as smart sensors cease to operate, and upgrades gracefully as new smart sensors are introduced in the system.

Dickson (1991) notes

> The correspondence between bayesian networks and Kalman filters should come as no surprise, as both are rigorous applications of Bayes' theorem, each in their particular domain. Whereas the Kalman filter gains its power and efficiency from the theory of linear transformation, no such simple mapping is available between the nodes of the Pearl network. The condition independence which is the centrepiece of bayesian networks is of more ancillary importance to the application (but not the theory) of Kalman filters.

Nicholson (1992) has explored to construct a predict–measure–update cycle, analogous to a Kalman filter, but liberated from the latter's dependence upon a linear algebraic representation of state and state change. The choice of belief networks is well-founded since the connection between them and Kalman filters has been established by Kenley (1986).

5. Purposive vision

Until quite recently the number of images that could be processed to yield information likely to be of use to a robot was quite small. For example, as recently as five years ago, it typically took an hour to process a stereo pair of images to produce a depth map. Now, with the assistance of any of a number of parallel distributed processors, the same computation may be completed ten times per second. Initially, this vastly increased processing power was used simply to execute previously slow algorithms more quickly. Aside from the enhanced understanding resulting from better visualization, the technology had little effect on the capabilities of mobile robots, just as when we watch a film we are powerless to alter its course.

However, it was soon realized that the increased speed of processing enabled the loop to be closed between sensing and action, so that a robot's behaviour could be modified according to what it had seen. Even though the design of visually guided control loops is difficult, and, as yet, poorly understood, from the standpoint of this meeting, of most significance are the gains that have already accrued from active or purposive vision. Simply stated this means that, on the basis of previous visual processing, the robot can act, not only to achieve a goal, but also to elicit additional useful visual information. We illustrate this idea with four examples.

It is well known that an important part of the early processing of an image by the human brain aims to isolate significant intensity changes. Intensity changes arise in several very different ways, including: a reflectance change, a lighting change (say, a shadow boundary), a sharp change in surface orientation, and a depth discontinuity (see Marr 1985 for more details). It has proved remarkably difficult to distinguish these different physical events on the basis of the intensity distribution across an intensity change. Recently, Cipolla & Blake (1992) showed that some differentiation is possible, and reliable, if the viewer makes a deliberate motion. More precisely, some intensity changes correspond to a physical contour that is fixed in space (e.g. a reflectance boundary or a sharp physical edge), whereas at others the surface normal turns smoothly away from the viewer. Cipolla & Blake showed that if a deliberate motion is made orthogonal to the viewing direction, these two cases can be distinguished reliably, and, in the latter case, the surface curvature normal to the bounding contour can be estimated. However prosaic this may seem, it has supported the construction of a path planner by which a robot can navigate among curved obstacles (Blake *et al.* 1991).

Building on the work referred to in the previous paragraph, Cipolla & Blake have also shown that a sequence of deliberate motions can enable a robot to determine its time-to-contact with a surface without explicitly computing the distance to it. More precisely, suppose that a planar contour (say a patch of a different colour to its surrounds) can be tracked as the viewer moves. There are a number of robust practical ways in which this can be done, even on modest hardware. Cipolla & Blake show that if a deliberate move is made orthogonal to the view direction, then the surface orientation of the patch can be determined. If a subse-

PHILOSOPHICAL TRANSACTIONS OF THE ROYAL SOCIETY A MATHEMATICAL, PHYSICAL & ENGINEERING SCIENCES

quent deliberate move is made along the line of sight, the time-to-contact can be computed from the divergence of the boundary of the patch (after correcting for the surface slant). Recently, Merron (1994) observed that similar considerations apply to vertical edges, so that the time-to-contact can be determined knowing only one's forward speed, and ignoring lateral and rotational motions. This is ideally suited for enabling a mobile robot to navigate about its environment.

A further implication of such work is to change in some cases the information that it is required to compute. Stereovision is a case in point. Conventionally, stereo algorithms operate in two stages. The first establishes correspondences between feature points in the two images, by determining that they arise from the same point in space. (A number of algorithms have been proposed for this stage, notably PMF (Pollard *et al.* 1986), which is fast, reliable and based on findings about human vision.) The correspondences established in the first stage each have an associated disparity, a measure of the change in image position from the left to the right view. The second stage converts disparity to depth. Roughly, depth is inversely proportional to disparity; but the exact relation depends on the careful calibration of the two cameras, a delicate, nonlinear task. Nevertheless, it has been assumed that the goal of stereo is to produce explicit measures of depth, despite the complexity and error introduced by the second stage. It now seems, as a result of recent work in purposive vision, that the second stage is unnecessary: the changing disparity field encodes most of what is needed to navigate about the environment. Working in a similar vein, Shapiro *et al.* (1994) have shown that one can compute the image of the instantaneous axis of rotation of a moving object, under certain quite weak assumptions.

Our final example concerns gaze (or attentional) control: where to look? It is well known that the human eye is an exquisite device that comprises two very different subsystems. A remarkably small fovea is dedicated to detailed shape analysis and colour processing, whereas the majority of the retina, though quite coarse in its spatial sampling, detects temporal changes very quickly. Such changes serve to direct gaze to parts of the scene that may pose a threat. The complex mechanism of attentional control is poorly understood (though see Carpenter 1988). Recently, robotics researchers have developed steerable camera platforms (Du & Brady 1994; Murray *et al.* 1992; Pahlavan & Eklundh 1990), not only to support experimentation, but to exploit the information that gaze control potentially provides. Threat detection aside, a system that can be made to track a target smoothly can build up a representation of its shape, perhaps even identify it, without being sidetracked by the complex problem of compensating for image motion (optic flow, retinal slip). Similarly, if a (stereo) pair of cameras both track an object, stereo matching can be made much easier because disparities can be kept small.

These are early days, but the prospect for AI is the increasing realization of systems that make deliberate motions to elicit information, either more robustly than previously, or which is otherwise not available, and build representations that enable goals to be achieved more efficiently and reliably.

6. Emergent properties

For almost one hundred years, scientific accounts of legged locomotion have been based around the concept of gait and duty cycle. Gait is the pattern of leg

PHILOSOPHICAL TRANSACTIONS OF THE ROYAL SOCIETY A MATHEMATICAL, PHYSICAL & ENGINEERING SCIENCES

motions and duty cycle refers to the relative amount of time that a leg spends on the ground as opposed to moving through the air to the next footfall. It has been noted, for example, that there are essentially three different gaits for a quadruped: the trot, the pace and the bound. Measurements of gait began with Muybridge's photographic sequences of animals in motion. In AI terms, gait has been the base representation for thinking about legged locomotion.

Ten years ago, Raibert (1986) challenged this assumption. He argued that the fundamental problem that a legged animal or machine has to solve is balance: legged locomotion is of little use if the system falls over and cannot locomote. From the standpoint (no pun intended) of engineering parsimony, the simplest system to start with is a monopod: an autonomous pogo stick. In that case, balance is key, gait is irrelevant, and static balance is impossible. Raibert built a monopod, initially constrained to move in a plane but soon liberated to hop anywhere on a planar surface.

For the purposes of this paper, the key step came in the subsequent development of a biped running machine. Once more, from an engineering standpoint, the biped could be constructed from two connected monopods. Crucially, the biped was controlled as if it were a virtual monopod, symmetrically placed between the two actual legs. The actual legs took turns to act out the role of the virtual leg, which moved in the sagittal plane; their alternation was used to null out extra-sagittal plane motions, a technical way of saying that they prevented the biped from falling over sideways.

How to construct a quadruped? Naturally, from two bipeds. But which? There are three choices: (i) the front pair and the back pair; (ii) the front right and left back, and the front left and right back; and (iii) the left pair and the right pair. In each case, each pair is controlled as a biped, hence as a virtual leg; and the two virtual legs are controlled as a pair of virtual legs to form a final virtual leg that simply hops forward. The three pairings correspond precisely to the three gaits. In short, gait arises as an emergent property of coupled oscillations of pairs of oscillating systems. Brooks's remarkable family of insect devices also demonstrate emergent behaviours from simple interacting behaviours. Remembering Lissajous' figures, never mind chaotic systems, we should not be surprised that complex patterns can be woven from the simple interactions between simple oscillating systems.

What does this have to do with the mind? Simon (1969) contrasted an observer's perception that the complex path followed by an ant moving across a beach must be the external manifestation of a massive intellect with the behaviour of a rather simple system confronted by a complex world. We have no doubt that the brain is immensely complex; but the point of the example is that we, as scientists, may over-complicate our theories by inferring complex mental apparatus from complex behaviour, when the complexity may reside primarily in the world. Agre (1992) has argued this point forcefully.

7. Conclusion

AI has for too long been dominated by reasoning and pure thought. Such work emphasizes static symbolic representations. In contrast, the world is forever changing and our behaviour in it is provided by our senses. Intelligence is always embodied, and the constraints that embodiment imposes through sensors and capabilities for independent motion are considerable. Work in robotics offers to the study of intelligence insights gleaned from building autonomous agents that operate purposefully in the real world, generating timely responses to situations, and contending all the while with uncertainty. This should come as no surprise: sensing and acting are what the brain evolved to be good at.

References

Agre, P. E. 1992 *The dynamic structure of everyday life*. Cambridge University Press.

Agre, P. E. & Rosenschein, S. (eds) 1995 Special issue. *Artificial Intelligence*. Amsterdam: Elsevier. (In the press.)

Bar Shalom, Y. & Fortmann, T. E. 1988 *Tracking and data association*. New York: Academic Press.

Blake, A., Brady, M., Cipolla, R., Xie, Z. & Zisserman, A. 1991 Visual navigation around curved obstacles. *Proc. IEEE Int. Conf. Robotics and Automation*, vol. 3, pp. 2490–2495.

Brady, M. 1985 Artificial intelligence and robotics. *Artificial Intelligence* **26**, 79–121.

Brooks, R. A. 1989 The whole iguana. In *Robotics science* (ed. M. E. Brady), pp. 432-456. Cambridge, Mass.: MIT Press.

Carpenter, R. H. S. 1988 *Movements of the eyes*. London: Pion.

Cipolla, R. & Blake, A. 1992 Surface orientation and time to contact from deformation of apparent contours. In *Proc. Euro. Conf. Computer Vision 1992* (ed. G. Sandini), pp. 187–202. Berlin: Springer Verlag.

Dickson, J. W. 1991 Image structure and model-based vision. D.Phil. thesis, Department of Engineering Science, Oxford University, U.K.

Du, F. & Brady, M. 1994 An active computer vision system for vehicle navigation. *Int. J. Pattern Recognition Artificial Intelligence*. (In the press.)

Fikes, R. E. & Nilsson, N. J. 1972 STRIPS: a new approach to the application of theorem proving to problem solving. *Artificial Intelligence* **3**.

Hoare, C. A. R. 1985 *Communicating sequential processes*. Englewood Cliffs, NJ: Prentice-Hall.

Hu, H. & Brady, M. 1994*a* A parallel processing architecture for sensor-based control of an intelligent mobile robot. *Robotics Autonomous Systems*. (Submitted.)

Hu, H. & Brady, M. 1994*b* Dynamic planning with uncertainty for an intelligent mobile robot. *IEEE Trans. Robotics Automation*. (Submitted.)

Kenley, C. R. 1986 Influence diagram models with continuous variables. Ph.D. thesis, Department of Engineering-Economic Systems, Stanford University, California, U.S.A.

Marr, D. 1985 *Vision*. San Francisco: W. H. Freeman.

Merron, J. A. S. 1994 Alternative depth representation for road-based vision. *Prometheus* report 5. Department of Engineering Science, Oxford University, U.K.

Murray, D. W., Du, F., McLauchlan, P. F., Reid, I. D., Sharkey, P. M. & Brady, M. 1992 Design of stereo heads. In *Active vision* (ed. A. Blake & A. L. Yuille), pp. 155–172. Cambridge, Mass.: MIT Press.

Nicholson, A. E. 1992 Monitoring discrete environments using dynamic belief networks. D.Phil. thesis, Department of Engineering Science, Oxford University, U.K.

Pahlavan, K. & Eklundh, J. O. 1990 A head-eye system for active, purposive computer vision. TRITA-NA-P9031, CVAP 80, Royal Institute of Technology, Stockholm, Sweden.

Pearl, J. 1988 *Probabilistic reasoning in intelligent systems*. San Mateo, California: Morgan Kaufmann.

Pollard, S. B., Porrill, J., Mayhew, J. E. W. & Frisby, J. P. 1986 Disparity gradient, Lipschitz continuity, and computing binocular correspondences. In *Proc Third Int. Symp. on Robotics Research* (ed. O. D. Faugeras & G. Giralt), pp. 19–26. Cambridge, Mass.: MIT Press.

MATHEMATICAL, PHYSICAL & ENGINEERING SCIENCES
THE ROYAL SOCIETY A
PHILOSOPHICAL TRANSACTIONS OF

Raibert, M. H. 1986 *Legged robots that balance.* Cambridge, Mass.: MIT Press.

Rao, B. S. Y., Durrant-Whyte, H. F. & Sheen, J. A. 1994 A fully decentralized multi-sensor system for tracking and surveillance. Oxford University Engineering Science Department report no. OUEL 1886/91. *Int. J. Rob. Res.* (In the press.)

Rosenschein, S. J. & Kaelbling, L. P. 1986 The synthesis of digital machines with provable epistemic properties. In *Theoretical aspects of reasoning about knowledge* (ed. J. Halpern). Morgan Kaufmann.

Shapiro, L. S., Zisserman, A. & Brady, M. 1994 Motion from point matches using affine epipolar geometry. *Int. J. Computer Vision.* (In the press.)

Simon, H. A. 1969 *The sciences of the artificial.* Cambridge, Mass.: MIT Press.

Discussion

B. Webber (*University of Pennsylvania, U.S.A.*). You've spoken of vision as if it were a single process, without linking it to behaviours.

M. Brady. I didn't want to give that impression. Vision should be thought of as the deliberate and active seeking of information. Sensing and planning must be dynamically interleaved.

MATHEMATICAL, PHYSICAL & ENGINEERING SCIENCES
PHILOSOPHICAL TRANSACTIONS OF THE ROYAL SOCIETY A