Robust Intelligent Control Through The Use of a Behaviour Based Control Paradigm

Martin D. Westhead

November 3, 1993

Abstract

This paper proposes a *behaviour based* strategy as a new approach to the problem of robust intelligent control in robotics. The classical paradigm is outlined and contrasted with a description of the new behaviour based paradigm. Finally an example system constructed at Edinburgh is described which uses hybrid control for assembly. It consists of a classical planner for high level planning connected to a series of reactive, behaviour based agents which control the robot movements. This control system has been demonstrated to be remarkably robust in performance.

1 Introduction

This paper considers the qualitative problem of robustness in an intelligent control system. It deals with the case of robot control, but the ideas could equally well be applied to the control of any complex system.

Producing reliable, repeatable behaviour from a robot is a very difficult problem. The difficulty lies in uncertainties that are inherent in interacting with the real world. There is at present no general way of dealing with these uncertainties.

Even in the case of assembly robotics where the environment can be engineered there still exists sufficient uncertainty to cause classical intelligent control systems significant problems. Most of the uncertainty here occurs in the transition from *robot* movements, which can be accurately determined, to *part* movements which may not occur exactly as predicted.

The difficulty of the problem has generally led to robot control programs being decomposed into functional subcomponents which are considered to be more tractable. These subunits are then built and tested independently in simulation. Unfortunately it has proved difficult to integrate these subunits into a complete system. At the end of this paper I present what is possibly the only complete assembly robot control system [MS88]. Complete that is, in that from a description of the shape of the finished product it is able to plan a gravitationally stable assembly of the parts and the necessary grips, and to reliably execute this plan. It is proposed that the robustness and reliability of the system, and the ease with which is was constructed are due to the behaviour based philosophy that was used in its design.

2 Classical Approach

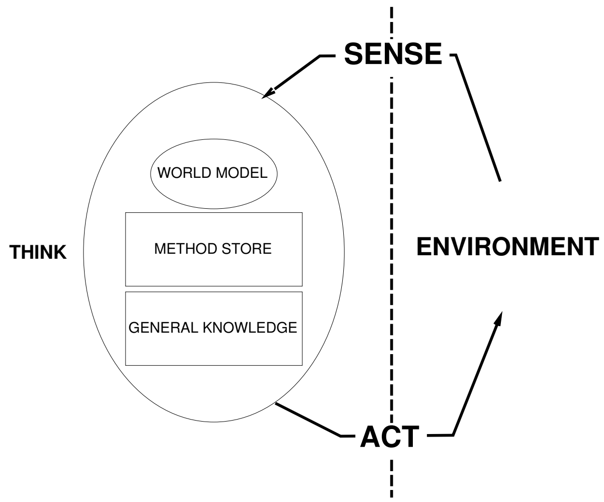

Figure 1: Sense-think-act cycle of the classical intelligent control paradigm.

In a classical system, intelligent control is typically achieved through the use of a large sense-think-act cycle (see figure 1). Control is decomposed by function. Typically a high level symbolic planner will produce a detailed complex plan which attempts to take into account all possible uncertainties and constrain them to within acceptable limits. Because these uncertainties can accumulate across a series of actions, such detailed plans are very difficult to generate.

An example of a working classical system would be the FMC Corporation's autonomous vehicle. The controller is composed of three major sub-systems for perception, planning and motion (sense, think, act). Given information about the mission, terrain type and vehicle characteristics and capabilities, the planning system generates a plan in the form of a sequence of tasks to be achieved which will satisfy the current goals. Segments of this plan are passed on to the perception sub-system. These plan segments specify desired behaviours. Directed by the requirements of these behaviours, sensory data is then collected and a local map built. This local map is fused with constraints generated by the plan, and passed on to the motion subsystem or pilot. The pilot then sends appropriate commands to the vehicle motors. The results of these actions are then fed back to the planning module.

The controller has been successfully used to drive an M113 armoured personnel carrier, following a planned route avoiding obstacles at 8km/hr, and following roads at 19 km/hr. The conclusion drawn from these results was that since the system clearly worked, the architecture was therefore suitable for real-time autonomous vehicle navigation.

3 Behaviour Based Approach

An alternative approach to intelligent control, is to decompose the problem in terms of the behaviour of the system (see figure 2). This is the approach taken by proponents of a new Paradigm in robotic control: Behaviour Based Systems. This approach is generally considered to be pioneered by the robotics group at MIT working under Brooks [Bro89]. Though there is increasing interest from many other workers throughout the world [MSH89].

Behaviour based systems are not constructed as a single large control loop, but rather a whole series of small, reactive control units which work together in parallel. The overall behaviour of the system occurs as a result of the combination of these interacting processes. It is said to be emergent.

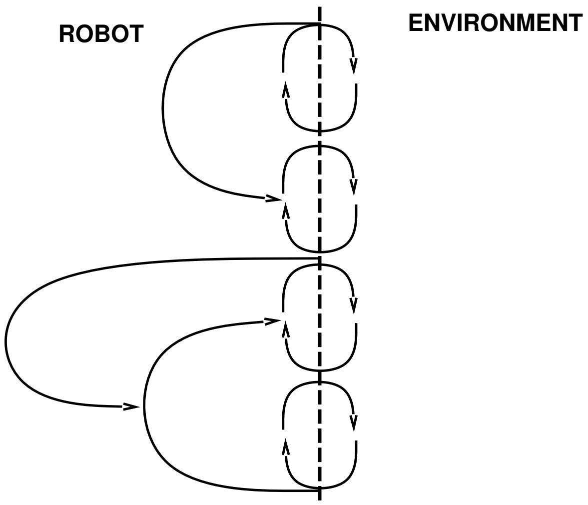

Figure 2: Illustrating the idea of low level amalgamation of sensing and action in a series of small, parallel behaviour based modules.

A good example which illustrates the principle is the robot Herbert [Con89]. Herbert collects drinks cans, lying around in a office environment. Herbert does not contain any high level representations of the real world. He consists, simply, of a series of independent reactive behaviours.

Herbert is equipped with a simple vision system, and will wander around his environment until he sees a doorway shaped structure. When he sees such a structure he will go through it and begin looking for table shaped objects, and then can shaped objects. If Herbert thinks he can identify a can shaped object on a table shaped object, he will approach it and stop. That's the end of that behaviour. Now at this point there are no messages sent, no representations updated. Herbert simply stops. Another behaviour controlling the arm notices the stillness and begins to reach out, lifting to get over obstacles, lowering to get near the surface. An orientation behaviour then notices the movement of the arm. It can see both arm and can in the picture and rotates the robot in such a way as to bring the two closer together. It is important to note however that the arm knows nothing about the rotation, and the rotation knows nothing about the movement of the arm. Each one is simply responding to the environment. When the hand gets close to the can a local sensor on the hand can detect its presence, guides it over the can, and grasps it. The arm behaviour is then triggered by the grasp to retract.

An important point to note here is the independence of these behaviours. Communication is achieved wherever possible through the real world. As Brook's describes it the real world is used to model itself. This gives the system a great deal of robustness since it does not suffer from the problem of discrepancies between its internal model and its environment, in the way that many classical systems do.

4 Comparison of the Approaches

The difference between these two approaches is founded in the way the control problem has been decomposed. It has been suggested [CG90, CG89] that there are three basic ways to parallelize a problem: in terms of an agenda of tasks, in terms of areas of specialization in the problem, or in terms of the final result.

An agenda decomposition is based around an agenda of tasks, which may have a partial ordering, used to feed a pool of generic workers. Such a decomposition is typified by the *task farm* programming model.

If we consider the construction of a house. In an agenda decomposition we would have a series of tasks: laying the foundations, building the walls, fitting the door frames, wiring the house, etc. The idle workers take tasks off the agenda, they complete the task and then return for another until the work is complete.

A specialist decomposition splits a task into functionally distinct components. In the case of a house we would consider the specializations of the workers. So we might have brick layers, carpenters, roofers, electricians, plumbers and so on. All the workers are started at the same time but some may have to block. For instance the electrician may have to wait until the roof is on the building before the wiring can be fitted. The parallelism in this model is often limited. It is often encountered as a *pipeline* where we would be building an estate of houses, so that the second house can be under construction while the electrician wires the first.

Finally, a result based decomposition looks at the desired result, and decomposes the problem in terms of its independent components. In the construction of our house these components are the roof, walls, windows doors etc. which could be prefabricated and then assembled. For a more detailed description of these "Paradigms of Parallelisation" see [Wes92].

The classical approach is a specialist decomposition. It decomposes the problem in terms of the specialist functions of the system leading to the pervasive sense-think-act cycle. One obvious consequence of this is that the potential parallelism is limited since each module must wait for the results from the previous module before it can proceed. In contrast the Behaviour Based approach is decomposes the task in terms of the result. The result of a robot control system is its behaviour, and systems that are constructed under this design philosophy decompose the problem in terms of (relatively) independent behavioural components which can act together in parallel

This new approach has yielded a number of apparent advantages over the classical architecture for robot control. There seems to be a significant saving in computational complexity. Brooks demonstrates this with his six legged walking robot [Bro89]. This has suggested to some that it is perhaps a more natural way to construct intelligent control systems.

5 A Robust Controller

The reason for the advantages of a behaviour based control system is to do with the way in which such systems deals with uncertainty. Because of the low level amalgamation of sensing and action provided by these reactive controllers, the complexities of reasoning about uncertainty are hidden from the planner. This can be used to achieve very robust control.

At Edinburgh a prototype assembly system has been built which embodies these principles. SOMASS [Mal91] is a hybrid system which uses a symbolic planner for reasoning about the construction of shapes in the *soma world*. The soma blocks were originally defined by the mathematician Piet Hin. They are a set of parts which provide a rich mathematical and combinatorial problem domain [Gar61]. They can be fitted together to form a number of shapes including a cube. This domain was chosen because it incorporates the feature of shape-dependent part-fitting which is an essential feature of assembly work.

The planner works in a perfect abstraction of the world to produce a plan in terms of *part* movements. The plan includes a details of the gripping strategies to be used by the robot. This plan is then passed to behaviour based agents which can reliably execute the plan and cope with real world uncertainty. The original system was developed in around six person months. In tests it was able to construct a number of different soma shapes, some for the first time with a success of around 98% [MS88]. The test represented around 45 hours of repetitive runs of some 4000 lines of automatically generated robot code. Most of the failures were due to trivial bugs in the planning system, and though this level of success is far from the standard required of an operating assembly system, it was seen as very encouraging. A further feature of this behavioural decomposition was that modules can be tested independently and so guarantees for the overall reliability of the system can be produced in advance of building it or without exhaustive testing.

The original system was developed without sensors. A later addition was a vision system to control part acquisition which allowed it greater freedom to cope with uncertainty. To illustrate the use of behaviours in the system I shall briefly describe the way in which this behaviour based vision system worked.

The way a robot vision system would usually work is to use a mathematical model to translate the coordinates of part as seen by a camera, into joint-space coordinates, so that the robot can move to that position and acquire the part. This requires very precise calibration of the cameras' position and orientation, and a very accurate model of the robots joints and the distances between them. All of this is expensive to keep accurate, and if, for example, a camera is knocked out of alignment, then the system will fail.

The SOMASS vision system [CC92] does not need any of this. A camera is mounted on a second robot arm, and nothing is known about its position or orientation with respect to the assembly robot, except that the camera knows how to point at a single rendezvous point.

The grippers rendezvous with the camera at this point, and then move off towards the location of the part. The assembly robot controlling the grippers knows the approximate location of the part to within a certain radius. The camera has a simple tracking behaviour. It can sense the movement of the grippers and follow them so that they are kept in the picture.

Once the grippers have reached the area they expect the part to be in they stop. Based on information from the camera image, the robot then moves its gripper in such a way as to try to reduce the distance from them to the part. The first move will be a guess based on what worked best last time however the perspective of the camera may well be different from last time so en error is expected. The robot can look at the change the gripper movement produced in the image, and use that to calculate a better move. In practice only a few attempts are required to produce a sufficiently accurate local model that the robot can reliably acquire the part. Once the part has been placed in the assembly, the robot can meet up with the camera again at the rendezvous point for the acquisition of the next part.

For complete generality two cameras are used since a single image is ambiguous with respect to movements along its line of sight. The principle remains unchanged however, the two cameras have no direct communication and the robot simply moves so as to minimize the distance between part and gripper on two images instead of just one.

This strategy is clearly far more robust than the classical approach. The camera can be knocked or even picked up and moved to a completely different place, the robot can be bent or moved, and the system will still work.

6 Conclusion

In conclusion the behaviour based approach appears to have a great deal of potential for the construction of robust intelligent controllers for complex systems. They are particularly effective in dealing with uncertainty which is one of the major problems with the control of any system which has to interact with the real world.

As yet this approach is still very new. There remains a lot of work to be done, particularly as to effective ways of designing these systems. It is not know how to best approach the problem of behavioural decomposition, nor how to design very complex behavioural controllers. Nonetheless the work described here shows that intelligent robust control of real robot systems can be effectively achieved using this new paradigm.

7 Acknowledgments

Many thanks to John Hallam, Chris Malcolm and Ashley Walker for there help in preparing this paper.

References

[Bro89] Rodney A. Brooks. A robot that walks; emergent behaviors from a carefully evolved network. A.I. Memo 1091, Massachusettes Institute of Technology Artificial Intelligence Laboratory, February 1989.

[CC92] Prabhas Chongstitvatana and Alistair Conkie. Active mobile stereo vision for robotic assembly. DAI Research Paper 590, Department of Artificial Intelligence University of Edinburgh, 1992.

[CG89] Carriero and David Gelernter. *How To Write Parallel Programs: A Guide for the Perplexed.* Scientific Computing Associates inc., 1989.

[CG90] Carriero and David Gelernter. *How To Write Parallel Programs: A First Course.* MIT Press, 1990.

[Con89] Jonathan H. Connell. A colony architectur e for an artificial creature. Technical Report 1151, MIT AI Lab, Aug ust 1989.

[Gar61] M. Gardener. *More Mathematical Puzzels and Diversions.* Penguin, 1961.

[Mal91] Chris Malcolm. Behavioural modules in robotic assembly. Teaching paper, March 1991.

[MS88] Chris Malcolm and Tim Smithers. Programming assembly robots in terms of task achieving behavioural modules: First experimental results. DAI Research Paper 410, Department of Artificial Intelligence University of Edinburgh, 1988.

[MSH89] Chris Malcolm, Tim Smithers, and John Hallam. An emerging paradigm in robot architecture. DAI Research Paper 447, Department of Artificial Intelligence University of Edinburgh, 1989.

[Wes92] Martin Westhead. Linda and the paradigms of parallelisation. IEE Coloquium: Control and Systems group C8 - Parallelisation in Control, February 1992.