Organisation and Design of Autonomous Systems

Chapter 1

Faculty of Mathematics, Computer Science, Physics and Astronomy (WINS)
University of Amsterdam
August 1999
Contents

1 Introduction Autonomous Systems ................................. 1
  1.1 Control Theory .................................................. 1
      1.1.1 Transducers .............................................. 7
      1.1.2 Control systems ......................................... 8
  1.2 Artificial Intelligence ........................................... 10
      1.2.1 Impact of AI on control ................................. 11
      1.2.2 Abstraction hierarchies ................................. 12
  1.3 Intelligent control .............................................. 13
      1.3.1 Knowledge-based control ................................. 14
      1.3.2 Fuzzy control ........................................... 15
      1.3.3 Neural network control ................................ 16
      1.3.4 Genetic control ......................................... 16
  1.4 Autonomous systems ............................................ 17
  1.5 Conclusions ...................................................... 18
  1.6 Literature ....................................................... 20
Chapter 1

Introduction Autonomous Systems

In this course autonomous systems will be discussed. An autonomous system is a system that is able to react on changing operating conditions and changing performance demands. The complexity of these systems lies in the fact that the design of these systems requires knowledge about different disciplines such as control theory, and AI formalisms such as neural networks and knowledge-based systems, stemming from completely different research fields. The combination of control theory and artificial intelligence is used to build an intelligent control system, a system which is intelligent enough to react on changing conditions. In this introduction we first will look more closely into control theory, thereafter discuss some AI techniques, and finally combine both approaches into an outline of autonomous control systems.

1.1 Control Theory

An autonomous system is part of the larger class of feedback control systems. An autonomous system, namely, observes its environment and attempts to take action when needed. But an autonomous system is a specific sort of feedback control system, because it can reason about the optimal control strategy. In control theory the word 'control' has the meaning of regulating, directing or commanding. Based on this notion the definition of a control system can be given:

Control system:
A control system is an arrangement of physical components connected or related in such a manner as to command, direct, or regulate itself or another system.

An example is a mirror that is controlled by a (stepping) motor to regulate a laser beam to specific positions. (See figure 1.1.) The motor is the
controller that controls the combination of the laser beam and the mirror. The total of motor, mirror and laser beam is called the control system. The motor is steered by setting a certain voltage to one of its input channels. The precise numerical value of this input for a certain desired position of the laser beam can be calculated from its relation to the angle of the mirror \( \theta \) and the location of the laser with respect to the mirror (i.e. angle \( \alpha \)).

![Diagram](image)

**Figure 1.1: Mirror via stepping motor controlling laser beam**

The example above illustrates that the model of a control system is quite complex, moreover it allows us to define two additional properties of a control system, being input and output.

**Input:**

The input is the stimulus or excitation applied to a control system from an external energy source, usually in order to produce a specified response from the control system.

**Output:**

The output is the actual response obtained from the control system. It may or may not be equal to the specified response implied by the input.

In our example the input is a certain voltage to the (stepping) motor and the position of a laser beam, the output is the movement of the laser beam to the actual position.

Control systems can be of three basic types; man-made, natural (including biological control systems) or both. In this course the control systems under study will be mainly of the first type. An example of the second kind
is presented in figure 1.2. A hand is controlled by the brain which activates the muscles. The output of this natural control system is the position of the hand which is observed or 'measured' continuously by the eyes. The observed position is compared with the desired position and the difference is used by the brain to direct the hand to a closer position near the object.

![Diagram of hand-eye control system](image)

Figure 1.2: Hand-eye control system

Control systems are classified into two general categories: open-loop and closed-loop systems.

Open-loop:
An open-loop control system is one in which the control action is independent of the output.

Closed-loop:
A closed-loop system is one in which the control action is somehow dependent on the output.

An example of an open-loop control system is the mirror control system. The position of the mirror is reached only based on the reference voltage to the motor. An example of a closed-loop control system is the hand-eye control system. The eye observes the position of the hand and produces a feedback signal to the brain that is combined with the input to control the muscles. Another example of a closed-loop control system is the heating system of an oven. The heater is switched on according to the measured
temperature, which is compared to the reference temperature. The closed-loop oven control system is an example of a simple control system, but there exist much more complex closed-loop control systems, like for example the human driving in a car. The control action performed by the driver is dependent on the actual automobile heading.

Because of the presence of a feedback loop closed-loop control systems are called feedback control systems. We have to study some of the general properties of feedback control systems before we are able to study autonomous systems more in detail.

Feedback:
The property of feedback in closed-loop systems permits the output (or some other controlled variable of the system) to be compared with the input to the system so that the appropriate control action may be formed as a function of input and output.

From this definition we see that the input to the controller of an open-loop system is the reference value, whereas the input to the controller of a closed-loop system is the difference between the reference value \( r \) and the feedback \( b \), often referred to as the error \( e = r - b \).

The way in which control systems are represented is in the form of block diagrams. In a block diagram we always make a distinction between the controlling system (the controller) and the controlled system (the process, controlled system). In case of our mirror example the motor is the controller, whereas the combination of the mirror and the laser beam is the controlled system or process.

**Figure 1.3: Standard block diagram of open loop**

**Figure 1.4: Standard block diagram of closed loop**

Standard block diagrams for open- respectively closed-loop systems are given in figure 1.3 and 1.4. In the closed-loop diagram we see a circle which
is the summing point between the reference input and the feedback. In an open-loop system the reference input $r$ and the error input $e$ are equal ($e = r - b$ with $b = 0$). In a closed-loop system the difference of the reference value and the measured value is input to the controlling system (controller, controlling system), which generates the appropriate input to the controlled system (process, controlled system). The physical quantity we want to control is called the controlled variable.

Figure 1.5: Block diagram representing motor control of mirror

Figure 1.6: Block diagram representing hand/eye control system

The block diagram of the example of the motor controlling the mirror is presented in figure 1.5. Remember this was an example of an open-loop control system. The output of the system, namely the actual position of the reflected beam, is not fed back to be compared with the voltage applied to the motor. The block diagrams of previously discussed closed-loop systems is presented in figure 1.6 until 1.8. The input to the control system in figure 1.7 is a desired or reference oven temperature, the output is the actual oven temperature, which is measured by the thermostat. The difference between the desired and observed temperature is used for controlling the heater. If the temperature drops below a set value, the heater is switched on, if the temperature rises above a set maximum it switches off again. In figure 1.8 a block diagram is given of the closed-loop system of the human who controls a car. The output of the system, automobile heading, is compared with the reference input, the road heading. The difference is input to the brain of the driver, which directs the hands so that the appropriate control action can be performed as a function of actual automobile heading and road heading.

A property common to all physical systems is that we are able to make a model for it, in which certain fundamental quantities can be defined by numerical values. The laws defining the relationships between these funda-
mental quantities are usually represented by equations based on the calculus of differences. With these differential equations it becomes possible to describe the behavior of a system in e.g. time (or frequency) or space.

The relation between the representative parameters of a given process $S$ is called a model of $S$. Some parameters can be set externally, which is the external input to the process $S$. The process $S$ is imperfectly known and disturbed by its environment. The strategy is to design a controller $S'$ that will set the control input of $u$ of $S$ on the basis of the current state and the external input. Mathematically these models are formulated as difference or differential equations. The main results of control theory have been obtained for systems described by linear difference or differential equations.

**Example:** To illustrate this we give the following example. Assume a helmsman is controlling the rudder of a ship which has to be maintained on a given course. This requires compensating disturbances to the course of the ship due to the wind, the waves and the currents. The problem is that the improperly known process $S$ (ship) relates the position of the rudder $u$ to the ships course $y$. By observing the actual course, and taking appropriate action to correct for deviations, the helmsman acts as controller $S'$.

As an example of a model of $S$ we can use the following equation for the relation between course and rudder:

$$\frac{d\theta}{dt} = c\phi$$

where $\theta$ is the course, $c$ a constant and $\phi$ the position of the rudder. The system $S'$ can be designed as a function of $\theta - \theta_{\text{desired}}$, where $\theta_{\text{desired}}$ is the new control input to the combined system. See figure 1.9.

This function may include terms with time derivatives, so a possible feedback law (model for $S'$) would be:

$$\phi = a(\theta - \theta_{\text{desired}}) + b\frac{d}{dt}(\theta - \theta_{\text{desired}}).$$
1.1. CONTROL THEORY

Control theory provides a methodology for the design of $S'$, in such a way that the resulting system has some desired properties. A major concern is stability, which means that after a disturbance has brought the system in a new state the system will return to the originally prescribed state. In addition to stability there are some other properties that may be determined, such as whether oscillations may occur before the system reaches its prescribed state, the maximum amount of overshoot, and so on.

Typically, the analysis of whether the system has obtained its desired properties does not take explicitly into account that there are finite bounds on the control input. The designer has some background knowledge on which area the system will operate in, how large the deviations are that may occur, and assures himself that the control inputs required by the feedback law are well below the maximum that can be given.

The implementation of $S'$ does not depend on the digital computer, for instance the steam engine designed by James Watt was controlled by a governor, a mechanical device. Before inexpensive microprocessors became available most control systems were implemented with analog electric devices, that implemented functions such as integration, addition, differentiation and multiplication. Since the designs of $S'$ made on the basis of control theory are simple continuous functions, such implementation technology is perfectly adequate. When designing a control system (find a controller which controls a process), the first step is to formulate a mathematical model of the system or process to be controlled. This model is expressed as a differential equation relating inputs (controls) of the system to output (response, observation), either directly or through the introduction of state variables. We will further work this out in chapter 3.

1.1.1 Transducers

Control systems almost always use devices that convert one form of energy into another. These devices are called transducers. Sensors belong to a specific class of transducers which produce an input signal to the control system. When transducers are used to realize a change in the control system or its environment, they are called actuators. Sensors are used as input (or feedback) transducers, actuators as output transducers. A block diagram

![Feedback control loop for helmsman controlling a ship](image-url)
of a standard feedback loop including the sensors and actuators is given in figure 1.10.

![Feedback control loop](image)

Figure 1.10: Feedback control loop including sensor and actuator

Examples of transducers are potentiometers that convert mechanical energy into electrical energy or a photo diode that transforms light into electrical energy.

Because a control system, whether open- or closed-loop, is always equipped with input and output transducers (sensors and actuators), it is also called a measuring and regulating system. Transducers are used to measure e.g. observe the environment and to actuate e.g. to regulate or control the environment. An example of an above described measuring and regulating system is a robot. Applications can be found in science in the form of automated measuring devices (a telescope, a spectrometer, an accelerator), or in industry in the form of automated machines or vehicles (an automated milling or drilling machine, a paint spraying robot, an unmanned vehicle on a shop floor). But also in mainly man-controlled systems like a car (the anti blocking system) or an airplane (numerous hydraulic control loops) these type of systems can be found.

### 1.1.2 Control systems

As was observed control systems use transducers to convert one form of energy in another. The oldest man-made control systems were of pure mechanical nature. A well known example is the mechanical weight controlling the pressure valve of a steam engine, invented by James Watt, preventing over-pressure in the engine. After the discovery of electricity, electronic devices took over as the implementation technique for control systems. First systems took the continuous output signal, also called analogue signals, as reference input and processed these signals via analogue control techniques that realized the control signal \( u \). Well known examples are the so-called servo systems. A servo system is a power-amplifier control system/feedback control system in which the output is a mechanical position or a time-derivative such as velocity or acceleration. Another example is the numerical controlled measuring device, such as a potentiometer which is controlled by a motor in figure 1.11.
1.1. CONTROL THEORY

The advantage of the electronic processing technique is that, in principle, there is no difference in processing analogue or digital signals provided that the information content in the signals is preserved. How this is done is the area of digital signal processing and is outside the scope of this course. The conversion from analogue to digital signals is done with an electronic device called Analogue to Digital Converter (ADC), the reverse is realized via a Digital to Analogue Converter (DAC). Using these devices it becomes possible to build a digital control algorithm performing the controller function. The ultimate and most flexible digital control device is a computer because it can be used to program any form of numeric equation.

However, to be useful as a controller the control signal $u$ has to be generated so fast in response on the output signal $y$ that the external device can be controlled in time. This property of real-time response in the feedback to external stimuli is essential, otherwise the result can be malfunctioning of the external device to be controlled which could have a disastrous effect.

Real-time computer:

A real-time computer system is a system which correctness of results is not only dependent on the correct result of the program performing the computation but also on the time for generating the result.

Figure 1.12 illustrates the role of a computer as real-time controller. The introduction of computers as a device to estimate function $L$ and use this knowledge in the controller, has given a tremendous technology push to feedback control systems. This even more because of the impressive computational speed of 100 MFlops and more of now a days RISC based compute engines. This gave the possibility for complex control functions to be
computed in real-time. Based on computerized techniques electronic filtering evolved from the Proportional-Integral Derivate (PID) control towards control based on more complex methods such as Kalman and Lyapunov techniques.

However, there always stayed the problem in the design of controllers, specially for more complex functions, that the correct control parameter values almost always have to be derived from previous experience and experimentation. With other words the designer of a controller, /indexcontroller, controlling system say an autopilot, has the phenomenological knowledge (heuristics) on the behavior of a certain domain (say automobile kinematics). Based on this experience, he knows how to tune the control parameters. Later in this introduction we will illustrate that it is exactly for this task that control theory had to look for Artificial Intelligence (AI) techniques.

It can be foreseen that systems with more and more sophistication will emerge in the future. This is even more the case in the area of complex engineering such as for the design of power systems, robotics in manufacturing and for the design of mobile platforms for unmanned missions in hostile or hazardous environments. The need for such complex feedback control systems will increasingly stress the possibilities of control theory to the limits, giving a push towards further exploring other possibilities such as the combination of techniques from control theory with those stemming from AI as mentioned before. As was already observed this is in particular the case when controller parameter tuning is needed but also when we have to involve different controllers for different practical situations.

There is second reason to combine control theory solutions with AI solution and that is the need for a human interface on the feedback control systems. Examples are the unmanned vehicles that are used to locate and unarm possible car bombs, or devices that have to operate in nuclear reactors, or unmanned planes that have to be guided to certain places. In these situations the system on one side has to perform certain feedback control function autonomously whereas on the other side it has to interact with a human being. The problem with the interaction is that it best can be done on some form of symbolic level because this is how humans interact with their environments. Also here we can observe a need to combine classical control methods with methodology stemming from other sources such as AI. In the next sections we will first introduce AI techniques, thereafter we will explain how these can be combined with classical control, to end with defining intelligent systems for autonomous control.

1.2 Artificial Intelligence

Unlike control theory AI research has always been more associated with the way biological systems, in particular humans, behave to solve complex prob-
1.2. ARTIFICIAL INTELLIGENCE

Problems. Since the late fifties AI has tried to develop intelligent machines. Since that time the AI community has associated thinking with symbolic reasoning. Thinking was modeled as a process of deriving new conclusions using known facts and rules. Only symbols are manipulated and numeric calculations are not considered relevant for intelligent behavior. Unless control theory that has developed a well established set of methods and tools based on difference and differential equations, AI became less involved with any technique in particular. Later in the seventies the interest of the AI community shifted towards knowledge representation and acquisition. Knowledge was acquired by interviewing experts and letting them describe their reasoning in terms of inferences. In the eighties the interest shifted to fuzzy logic in order to reason with uncertainties, whereas in the nineties an addition came via a renewed interest in learning via neural networks and genetic programming.

1.2.1 Impact of AI on control

The influence of AI techniques on control systems was one of the driving forces behind what first became adaptive control and later autonomous control. In order to better understand the role of AI techniques in the design of control systems it is good to look back into what we have learned on control theory.

It was illustrated that the aim in the design of a controller, controlling a certain process is to make the difference between the desired outcome of the process and the actual behavior of the process as small as possible. An example was the helmsman on a ship whose task it was to compensate for the disturbances caused by the rudder, the current and the wind, in order to keep the difference between the desired ship course and its actual course as small as possible. The way he is doing this is, is by applying small corrections when the ship is in danger of leaving its defined course. If an autopilot, as a controller, takes over this task from the helmsman it is precisely performing these activities. To be able to do so the controller uses a model of ship and of the environment. Based on the measurements of some parameters of the model, say the current, it constantly updates the parameters. In order to prevent too drastic changes in the parameters and consequently instabilities in the control loop, the designer puts predefined limits on the amount of changes the parameters can undergo per control step. When the system control acts within those foreseen limits the feedback is defined as nominal. However, there could be a disturbance demanding more drastic parameter changes. For instance, because another large ship is crossing the foreseen ship course that might lead to a collision. In such a situation the helmsman will react because he will observe that the actual ships course will result in this collision. Consequently he will overrule his current control actions possibly applied by a new controller that allows a control model
with far larger changes between steps leading to a drastic other course. To
do so he will call for another capability that apparently is built in human
intelligence. It could be described as reasoning, making the inference that
his actual path planning would result in a disaster, and that consequently a
drastic new path had to be planned. Translating this to our controller would
imply that the current feedback values cannot be called nominal anymore,
because a continuation of the current control function will 5 to a collision.
The feedback values are now called non-nominal. As soon as this situation
is recognized, it has to be handled. A different control function has to be
activated, and for the selection of this function 'artificial' intelligence is
needed.

This could be realized either to let another process control the param-
ters and remove the limits or by introducing a completely different feedback
process provided by another controller. In both cases the target is collision
avoidance. In both cases we need a higher level process that is able to over-
rule the current situation. Such a process is clearly supervising the other
control processes and is consequently at a higher level of abstraction.

Processes operating on different levels of abstraction is an issue that
almost always occurs when intelligent control is applied in autonomous sys-
tems.

1.2.2 Abstraction hierarchies

The application of different layers of abstraction is a technique often applied
in computer science. It is done to separate different problems specially when
complex systems with a large number of hardware- en software components
have to be constructed. It allows to concentrate only on those specific prob-
lems one wants to tackle and that are characteristic for a particular layer.
Moreover, it aims to construct a system with different hierarchical layers.
One of the most well known examples is the virtual machine layer concept
in computer architecture. In this concept the micro programming level is
using machine micro instructions (the lowest level of abstraction), which are
generated from assembler instructions (one level higher abstraction). These
are generated from instructions of a higher level language (the next higher
level), which on their term can be obtained from a database or a traffic con-
tral system (the highest abstraction level). Two observations can be made
for this particular example:

1. Because we are dealing with virtual machines of increasing hierarchi-

al complexity, a higher level process is always in control of a lower
level process. It can be translated into the lower level process either
via compilation techniques or by using an interpreter. In our exam-
ple the database is translated into higher level language instructions
which are then translated in machine and in assembly instructions.
1.3. INTELLIGENT CONTROL

This hierarchy of levels of increased complexity and functionality with the top-level in charge is characteristic for hierarchical decomposition creating different abstraction levels of increased functionality. This is not necessarily always the case, nor is it always necessary that all functions of a higher level are translated into the next lower level. Sometimes levels are skipped. The bottom line is always that there is a lowest level that is responsible for the execution and that the highest level has control.

2. Different hierarchical layers are applied in order to provide information hiding and data abstraction. This allows the user of the system, either being an expert or a pure operator, only to concentrate on the problems he is interested in, or trained for. In our example the database expert will only concentrate his activities such as creating or improving performance of a database system. The micro programmer will concentrate on how to define optimal assembler instructions and getting the last horsepower from his machine.

Applying these ideas to AI the observation has to be made that because of the target of AI has been to reason, to learn or to capture knowledge, the research has concentrated on language methods and models that allow a sufficient high level of abstraction necessary to manipulate symbols. This has resulted in the concentration on so-called symbolic languages such as Lisp and Prolog. These languages were developed targeting to give the user maximum symbolic expressing power, so that he could concentrate on his particular problems, being AI research.

1.3 Intelligent control

Intelligent control is the discipline in which control algorithms and control systems are developed by emulating certain characteristics of intelligent biological systems. As a consequence it is a discipline combining control theory and algorithms with AI methods and techniques. One of the key issues in applying this type of AI methodology is whether the resulting control algorithm can be verified either by modeling and simulation, or by non-linear analysis, or by pure experimentation, as is done for conventional control systems. The advantage of intelligent control is the fact that AI based methods can be applied to conventional systems for, for instance, steering, braking or throttle control. Such it becomes possible to automate humans responsibilities when driving over highways. As we have observed there is a large amount of domain knowledge involved when trying to tune control parameters for a particular task. In addition similar knowledge is needed for the decision between different controllers suitable to tackle different control situations.
AI methods can be used to facilitate the formulation and acquisition of domain knowledge. Fuzzy control and knowledge-based control techniques can be used to facilitate the formulation of the heuristic rules by the human designer. Neural network techniques can be used to learn the heuristics by the system itself, which requires that the human designer formulates how the heuristics has to be learned. AI methodology can be helpful in those situations and therefore will be further discussed here. Broadly speaking four different classes of intelligent control are being considered today. They are classified on the basis of the respective application of AI methodology under consideration:

1. Knowledge-based control
2. Fuzzy control
3. Neural network control
4. Genetic control

This introduction will be finished by shortly describing these control methods and providing an example when possible. Because knowledge-based control will be extensively treated in the rest of this course, the main emphasis will be on the other methods.

1.3.1 Knowledge-based control

Knowledge-based control systems like expert systems have been applied to supervise controllers for complex industrial applications. The task in these case is more to supervise between different controllers and apply that controller optimal for a particular task (see figure 1.13). In robotics, planning systems are being used to perform complex path planning. In these situations the planner generates the appropriate parameter settings to the robot controller(s) to perform complicated paths that are necessary for the robot tasks.

Figure 1.13: Knowledge-based control, the rule-based supervisor switches between 1 kind of controllers, the diagonal arrow indicates that the controller is being tuned
1.3.2 Fuzzy control

One of the most widely published techniques for embodying human-like thinking into control system is fuzzy control. It is claimed that a fuzzy controller can be designed roughly to emulate the human deductive process, the process humans use to infer conclusions from what they know. In a fuzzy controller of which an example is illustrated in figure 1.14 four main components can be distinguished: a rule base, a fuzzification interface, as well as an input- and output fuzzification interface. The rule base holds the set of IF-THEN rules that quantify the knowledge human experts have amassed about solving the particular problem at hand. It is the input to the fuzzy inference mechanism, which makes successive decisions about rules most relevant to the current status and applies actions indicated by those rules. The interfaces translate numeric into symbolic information and vice versa.

Figure 1.14: Fuzzy cart distance control system, the system continuously adjusts the throttle $u(t)$, in an attempt to minimize the difference between the desired vehicle spacing $r(t)$ and the actual spacing $y(t)$

The design objective is to emulate the human behavior to regulate the inter-vehicle distance and keep it constant.

Let $e(t)$ be the error between desired and actual inter-vehicle spacing and $u(t)$ being the throttle input. Then the IF THEN rules would be of the following type:

![Fuzzy controller diagram](image-url)
IF $e(t)$ is positive-small and $de(t)/d(t)$ is positive-medium THEN
$u(t)$ is positive-medium

IF $e(t)$ is positive-small and $de(t)/d(t)$ is negative-medium THEN
$u(t)$ is positive-small

The first rule quantifies the drivers knowledge that the error between the desired spacing (reference input $r(t)$) and the actual spacing $y(t)$ is neglectable small and that we should continue with the medium throttle input. The second rule indicates that when the error is small but decreasing at medium rate, a smaller throttle input $u(t)$ should be applied. The danger of applying these rules straight towards the control system is that they will result in an input $u(t)$ that with each new control step is driven to its limits (from positive to negative). This form of rule-based control is often called bang-bang control. The problem with this control is that in the neighborhood of $de(t)/d(t) = 0$ the throttle input is switching between medium and small values, which means the throttle is opened and completely taken back again continuously. This can be solved by adding smoothing functions to the defuzzing interfaces. Often it is necessary to apply two different controllers, one for small movements and one for large. This is independent of the type of AI-control that is 1 and illustrated in figure 1.13. A /index*supervisor is then required to decide when to switch from one controller to another. In the example of an autopilot controller for a ship, the controller allowing large movements can be applied when the planning systems discovers a possible collision course and a large variation from the current course has to be applied for collision avoidance.

1.3.3 Neural network control

Next to fuzzy control neural network control has stirred much interest in the recent years. Similar to fuzzy control it is tried to emulate the biological functions of the brain to solve difficult control problems. Similar to the other application of AI techniques validation is an important issue. After training the neural network could be applied to the inter-vehicle regulation problem by recalling the proper throttle value for each value of the inter-vehicle distance that is sensed. Other neural network approaches bear similarities with the fuzzy logic approaches discussed before. The application of neural networks for vehicle control will further be discussed later in this course.

1.3.4 Genetic control

The area of genetic algorithms is young when its application to intelligent control is concerned. In genetic algorithms the target is to embody the principles of evolution, natural selection, and genetics from natural biological systems. Genetic algorithms perform a parallel stochastic but directed search to evolve the most fit population. It has been (off-line) illustrated
that genetic algorithms can artificially evolve an appropriate controller. To do so genetic algorithms maintain populated strings, each representing a different controller (digits on the strings characterize parameters of different controllers). The algorithm operates on those strings with genetic operators like crossover and mutation (representing among others survival of the fittest) coupled with a fitness measure (which often involves a performance objective) to spawn successive generations of population. After many generations the algorithm produces an adequate solution to the control problem. Its stochastic, but directed search, avoids local optima. A recent more successful approach is to make controllers that evolve while the system is actually operating. A reference model is used to characterize the desired performance. The key is an algorithm that maintains a population of strings representing candidate controllers. The algorithm employs a model of the process, along with the process data to evaluate the fitness of the controllers at each time step. Based on this evaluation the algorithm propagates controllers into the next generation using three standard genetic operators. In the end that controller what deemed most fit is used. This gives the possibility of controller parameter tuning. An impressive result is presented in figure 1.15 where this system is used in simulation to control a cargo ship. The reference input \( r(t) \) is the desired heading, the reference model output \( y_m(t) \) is the practical desired heading, the output \( y(t) \) is the actual heading and the control input \( u(t) \) is the commanded rudder angle.

\section*{1.4 Autonomous systems}

The ultimate goal of autonomous systems is a system that performs well under a large variety of operating conditions and performance demands. One way to realize this is the application of intelligent control. We have seen that the combination of control theory and AI techniques provides a combination to tackle the wide variety of problems to be overcome. A good architecture model is a first step for integrating these methodologies stemming from such different research communities. The basis for such a model is derived by applying methods of abstraction hierarchies as described in the section 1.2.2. Therefore, intelligent controllers are designed using distinguishing different levels of abstraction. The choice of the levels is rather arbitrarily. An illustration of a three layer system will shortly be introduced. Later in this course systems with more or other levels of abstraction will be discussed in more detail. The hierarchical control division we will look at here is one between an execution layer, a co-ordination layer and a management layer. (See figure 1.16.) The execution layer has the task to connect to the process under control via the sensors and actuators. The management layer interfaces with other systems and with the human operator(s). The exchange of information between these layers is done via the co-ordination layer. Each
of the layers has a considerable degree of autonomy. The execution layer is not just collecting data, it also performs control algorithms like PID control, parameter estimation and failure detection. The co-ordination layer tunes, schedules, supervises the execution control algorithms. It also handles crisis management, planning and learning. The management layer supervises the lower level functions and manages the interfaces between humans. In particular, it interacts with the users in order to define and generate the actual goals for the controller etc. This gives some impression of the different functionalities that have to be present in modern intelligent control for autonomous systems. Further details will be discussed in the next chapters.

1.5 Conclusions

In this chapter the basic ideas from control theory were introduced. The definition of a controller and a process as well as of a control system were presented. It was illustrated that control engineering is using block diagrams in order to analyze and design complex control systems. In addition it was explained that the basic ideas from control theory in designing a control
Figure 1.16: A three level autonomous system
control theory with AI technology to realize autonomous control. As the previously mentioned fields use completely different models and paradigms (the first numerical, the second symbolic techniques) it becomes necessary to apply sound architecture models to combine them into autonomous control systems. Abstraction hierarchies is an architecture model that is successfully been applied in information technology to solve complex system design problems. It was illustrated that this technique can be used when designing a autonomous controller for complex control problems. Because of the property of data abstraction and information hiding it is a good methodology for designing complex information technology based systems of which autonomous control is an example. We will now study autonomous systems, their underlying principles as well as the way in which they are designed in more details.

1.6 Literature

Index

abstraction, 12, 13, 17, 20
abstraction level, 12, 13
actuator, 7, 8
adaptable, 11
analysis, 7, 13
architecture model, 17, 20
artificial intelligence, 1, 10
AI formalisms, 1
AI methodology, 13
AI techniques, 10, 11, 13, 17
AI technology, 20
autonomous control, 11, 20
autonomous system, 1, 17, 20
control engineering, 18
control input, 6, 17
control parameter, 10, 13
control signal, 8, 9
control system, 1, 2–5, 7, 8, 11, 13, 16, 18
autonomous control system, 1, 19, 20
closed-loop (feedback) control system, 3, 4, 5, 19
conventional control system, 13
feedback control system, 1, 4, 8–10
intelligent control system, 1
open-loop (feedforward) control system, 4, 5, 19
open-loop control (feedforward) system, 3
control theory, 1, 6, 10, 11, 13, 17–19
 bang-bang technique, 16
PID technique, 10
to controlled variable, 4, 5
controller parameter, 10
controller, controlling system, 2, 4–7, 9, 10, 18
current state, 6
data abstraction, 13, 20
differential equation, 6, 7, 11
error input, 5

expert system, 14
external input, 6

feedback, 4, 7, 9, 10, 12
nominal feedback, 11
non-nominal feedback, 12
feedback law, 6, 7
feedback loop, 4
feedback signal, 3
fitness measure, 17
fuzzy control, 14, 15
fuzzy logic, 11
genetic algorithms, 16
genetic control, 14, 16
genetic programming, 11
heuristics, 10, 14
hierarchical layer, 12, 13

information hiding, 13, 20
input of control systems, 2
intelligent autonomous system, 10
intelligent behavior, 11
intelligent control, 13, 17
interaction, 10
Kalman filter, 10
knowledge representation & acquisition, 11
knowledge-based control, 14
knowledge-based system, 1
mission, 10
modeling and simulation, 13
neural network, 1, 11, 14, 16
operating conditions, 1, 17
output of control systems, 2
output signal, 8, 9
performance demand, 1, 17
process, controlled system, 4, 5, 7, 18
real-time computer, 9
real-time response, 9
reasoning, 12
reference, 4
reference input, 5, 17
reference output, 17
representation, 11

servo system, 8
state variables, 7
symbolic reasoning, 11

task, 14
transducer, 7, 8

verification and validation, 13, 16
virtual machine, 12
virtual machine layer, 12