Appendix A
Control theory

Angelo Coppola[1] and Roberta Di Pace[2]

The goal of the scientist is to comprehend the phenomena of the universe he observes around him.

Richard Bellman

Outline. *This appendix delves into the fundamental aspects of automation control, with a particular emphasis on its application in traffic control. As the complexities of modern transportation systems continue to grow, the integration of automation control becomes increasingly critical to enhance efficiency, safety, and sustainability. Accordingly, this section aims to provide readers with a solid foundation in automation control principles, tailored to the unique demands and challenges of traffic systems.*

A.1 Introduction to automatic control

Automatic control, often referred to simply as control, refers to the field of engineering and applied science concerned with the design and implementation of systems capable of autonomously regulating processes to achieve desired performance. It encompasses both theoretical methodologies and practical techniques that enable technical systems to execute predefined tasks with minimal human intervention. In other words, its primary objective is to regulate processes to achieve desired behaviors or performance outcomes.

Automatic control systems are typically composed of different components, as depicted in Figure A.1.

The process/system is the dynamical system that needs to be controlled (e.g., traffic flow in an extra-urban network). Note that a dynamical system is a system that evolves over time; at any given time, a dynamical system has a state representing a point in an appropriate state space. This state is often given by a tuple of real numbers or by a vector in a geometrical manifold. The evolution rule of the dynamical system, known as the dynamics of the system, is a function that describes what future states follow from the current state. The dynamic of the system depends upon:

[1]Department of Civil, Architectural and Environmental Engineering, University of Naples ‘Federico II’, Italy
[2]Department of Civil Engineering, University of Salerno, Italy

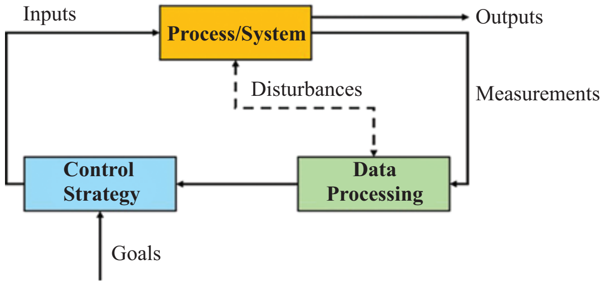

Figure A.1 Overview of the fundamental components of an automatic control system

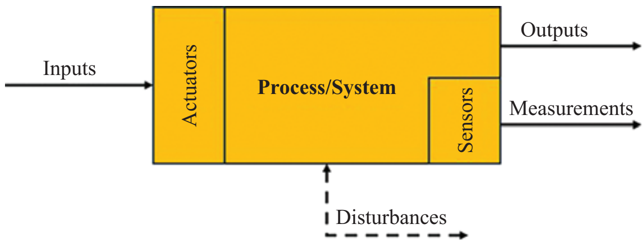

Figure A.2 Detail of the Process/System block

- External excitations (or quantities), presumed to remain unaffected by the dynamic progression of the process. In the case of extra-urban traffic, the external quantities can be exemplified by traffic demand, origin-destination patterns, and environmental conditions.
- The inherent behaviour of the process is due to its technical and physical attributes, illustrated by factors such as travel times or the accumulation of vehicles in a queue on an extra-urban link, as well as the flow capacity.

The process/system requires and includes two sub-components, as depicted in Figure A.2.

- Actuators: a set of instruments that allow acting on the process, affecting its behaviour or output. With specific regard to the traffic field, examples are panels that allow for variable messages.
- Sensors: instruments that measure the output or state of the process and convert it into a form that can be used by the control system. The information obtained from the sensor is called feedback.

The process/system is characterised by variables that enter the process/system itself or exit from it.

The variables entering the process are external excitations that can be categorised as follows:

- Input variables, come from an acceptable control region to regulate/control the process/system; an input variable can be exemplified by components such as variable speed limits.
- Disturbances are exogenous variables that may be estimated or predicted through appropriate algorithms but cannot be controlled.

The variables that come out of the process are usually referred to as output variables. The process outputs are the quantities chosen to represent the behavioural aspects of interest (e.g., the outputs of extra-urban traffic may be the average speed or density). However, not all the output variables of the process can be measured. As depicted in Figure A.1, it is possible to distinguish between the generic set of process outputs and measurements, indicating the sub-set of measurable outputs.

Real-time measurements of the process outputs and, eventually, disturbances are exploited by the data processing block to perform estimation and/or prediction tasks. The data processing block has a crucial role since it processes the received data in a way that makes it usable/useful for the control strategy.

Finally, the control strategy block performs the following task:

> *To specify in real time the process inputs, based on available measurements/estimations/predictions, so as to achieve the prespecified goals regarding the process outputs despite the influence of various disturbances.*
>
> Papageorgiou (1998)

The control strategy block is composed of an algorithm that takes the feedback from the data processing block and compares it with a reference or desired value. Based on this comparison, the algorithm generates a control signal (input signal) that is sent to the process. The control algorithm can range from a very simple control law to highly sophisticated control strategies and must follow given control requirements related to the goals and the characteristics of the controller itself. Examples of control requirements are the time to compute the control signal, a specific desired behaviour, or a given performance level of the process.

A.2 Process and system models

The process/system block depicted in Figure A.3 can be either real (physical) or a mathematical model describing the functional behaviour of a real entity. In this case, a set of equations describes, with varying degrees of precision, the intrinsic (internal) behaviour of the process/system within the specified context. The level of complexity, resolution, and accuracy of a mathematical model can vary, and different models may be employed based on the specific requirements of the analysis or simulation. Consequently, a mathematical model, when provided with input and disturbance parameters, can be utilised to compute, within a defined level of precision, the corresponding output values or other pertinent internal variables. To illustrate, an extra-urban traffic flow model, furnished

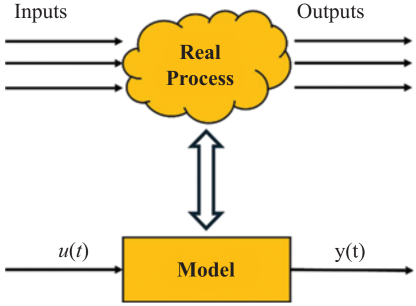

Figure A.3 Schematic representation of a real process and its corresponding model

with data pertaining to traffic demand, origin-destination patterns, and vehicle speed can be employed to ascertain queue formation/lengths and travel times across all network links.

In system theory, the functional relationship between inputs, models, and outputs is represented graphically by means of a flow chart, as Figure A.3 shows.

In the following, $u(t) = [u_1(t), \ldots, u_m(t)] \in \mathbb{R}^m$ will indicate the input vectors, $y(t) = [y_1(t), \ldots, y_p(t)] \in \mathbb{R}^p$ the outputs vector, $t \in \mathbb{R}$ the time variable.

Mathematical models can be classified into several categories based on their characteristics, purposes, and applications:

- Nature of the model
  - Physical models: Derived from physical laws that govern the system's behavior.
  - Empirical models: Developed from experimental data and observations, without relying on the underlying physical laws.
- Model behaviour
  - A model is static or algebraic if its output $y(t)$ depends only on the actual value of the input $u(t)$. These models do not take into account inputs at previous time instants, so do not give any information about the regime transient, i.e., on the time trend of the outputs during the transition from one steady state to another. The functional behaviour is described by an algebraic equation.
  - A model is dynamic if its output $y(t)$ depends on actual and previous values of the inputs.
- Signal types
  - Continuous-Time models (CT): Represent systems where the state changes continuously over time. These models are described by differential equations.
  - Discrete-Time models (DT): Represent systems where the state changes at discrete time intervals. They are represented by a set of finite-difference equations.
- Input-Output relationship
  - Linear models: Assume a linear relationship between input and output. Superposition and homogeneity principles apply.
  - Nonlinear models: Involve nonlinear relationships between inputs and outputs.

- Time dependency
  - Time-Invariant models: The system's behaviour and parameters do not change over time, and the relationship between input and output remains constant.
  - Time-Variant models: The system's dynamics or parameters change over time.
- Determinism
  - Deterministic models: The signals of a deterministic model can be described by analytical relationships.
  - Stochastic models: The signals can be given by probabilistic variables and/or contain randomness and uncertainties in the system's dynamics, inputs, or parameters.
- Number of input and output variables
  - Single Input Single Output (SISO), if $u(t) \in \mathbb{R}$ and $y(t) \in \mathbb{R}$ are scalar
  - Multi-Input Single Output (MISO), if $y(t) \in \mathbb{R}$ is scalar, but $u(t) \in \mathbb{R}^{m\times 1}$ is not
  - Single Input Multi Output (SIMO) if $u(t) \in \mathbb{R}$ is scalar, but $y(t) \in \mathbb{R}^{p\times 1}$ is not
  - Otherwise it is called Multiple Inputs Multiple Outputs (MIMO)

A.3 Dynamical models

A *dynamical system* is a system that evolves in time according to a predefined behaviour. In other words, the system interacts with the surrounding environment and, accordingly, evolves over time by means of some causes (inputs) operating on it which determines some effects (outputs), that are the response of the system to such stimulations. The knowledge of the input at time t is not enough to univocally determine the value of the output at time t. Indeed, the output $y(t)$ depends on the actual and previous values of the inputs. The functional behaviour is described by one or more differential equations that express static links between the input and output variables and their derivatives compared to time. In so doing, the dynamic model allows us to determine the trend of the output signal corresponding to a given input signal, i.e., to determine the response of the system to a given excitation.

To characterise a dynamical system, it is mandatory to define the 'specific object' (which part of the system) that evolves over time and the 'rules' that specify the evolution behaviour of the object. In so doing, a dynamical system becomes simply a model describing the temporal evolution of a system.

The first step is to define the set of variables that give a complete description of the system at any specific time, i.e., the variables must completely summarise the past history of the mathematical system. These variables are called *state variables*, while the set of all the possible values of the state variables is the *state space*. The number of state variables n is the *order* of the system/process. The state space can be discrete, consisting of isolated points, such as if the state variables could only take on integer values. It could be continuous, consisting of a smooth set of points, such as if the state variables could take on any real value.

Definition. The *state* $x(t_0) = [x_1(t_0), \ldots, x_n(t_0)] \in \mathbb{R}^n$ of a process/system in a generic time instant $t_0 \in \mathbb{R}$, assumed as the *initial instant*, is the minimum information necessary to uniquely determine the output $y(t)$, $\forall\, t \geq t_0$, once assigned the input $u(t)$.

The second step is to specify the rule for the time evolution of the dynamical system. This rule must be such that the value of the state variables at a particular time must completely determine the evolution of all future states. The time evolution rule will be based on a function f that takes as its input the state of the system at one time and gives as its output the state of the system at the next time. The time evolution rule could involve discrete or continuous time. If the time is discrete, then the system evolves in time steps, and usually integer time points are used $t = 0,1,2,\ldots$. In this case, starting from the initial conditions $x0$ at time $t = 0$, it is possible to apply the function f once to determine the state $x1 = f(x0)$ at time $t = 1$. If one applies the function repeatedly, it is possible to determine all future states, obtaining a trajectory of the points $x0$, $x1$,$x2$,$x3$,.... If the time is continuous, the state $x(t)$ at time t can be thought of as a point that moves smoothly through the state space. In this case, starting with an initial state $x(0)$ at time $t = 0$, the trajectory of all future times $x(t)$ will be a curve through state space.

Putting together what was said above, it is possible to describe a generic dynamical system using the state-space representation, which is a system of two vectorial equations that fully characterise the system's configuration at any point in time:

$$\dot{x}(t) = f(x(t), u(t), t),\ x(t_0) = x_0 \tag{A.1}$$

$$y(t) = \eta(x(t), u(t), t) \tag{A.2}$$

if the time is continuous;

$$x(k+1) = f(x(k), u(k), k),\ x(k_0) = x_0 \tag{A.3}$$

$$y(k) = \eta(x(k),\ u(k), k) \tag{A.4}$$

if the time is discrete.

The first equation, named *state equation*, characterises the evolution of the state $x(t)$, $\forall\ t \geq t_0$, given the initial state $x(t_0)$ and the input $u(t)$. The second equation, named *output transformation*, defines the output $y(t)$, at the generic time instant t, given the state $x(t)$ and the input $u(t)$. The terms $\dot{x}(t)$ and $x(k+1)$ are the derivatives of the state vector with respect to time, in continuous and discrete time respectively, indicating how the state changes over time. The function f, named *transition* or *state function*, maps the system's condition from its initial state and time to subsequent states, capturing how the system evolves. It is represented as a vector-valued function in the case of continuous dynamical systems or as a sequence in discrete systems. The function g, named *output transformation*, maps the current state and input to an output.

A.3.1 Continuous time systems

Here the properties of continuous-time systems are summarised and discussed.

Let us start considering a system of n first-order linear ordinary differential equations (ODEs) with inputs:

$$\begin{cases} \dot{x}_1(t) & = & a_{11}x_1(t) + \ldots + a_{1n}x_n(t) + b_1u(t) \\ \dot{x}_2(t) & = & a_{21}x_1(t) + \ldots + a_{2n}x_n(t) + b_1u(t) \\ \vdots & \vdots & \vdots \\ \dot{x}_n(t) & = & a_{n1}x_1(t) + \ldots + a_{nn}x_n(t) + b_nu(t) \\ x_1(0) & = x_{10}, & \ldots \quad x_n(0) = x_{n0} \end{cases} \qquad \left[\dot{x} = \frac{dx}{dt}\right]$$

where $x = [x_1 \ \ldots \ x_n]' \in \mathbb{R}^n$ is the state of the system of order n. The equivalent matrix form of the linear ODE system is the so-called linear system:

$$\dot{x}(t) \ = \ Ax(t) + Bu(t) \tag{A.5}$$

with initial condition $x(0) = x_0$, with vector $x_0 = [x_{1,0} \ \ldots \ x_{n,0}]' \in \mathbb{R}^n$.

The nth-order ODE can be recast in a more compact notation as a 1st-order linear system of ODEs:

$$\begin{aligned} \dot{x}(t) \ &= \ Ax(t) + Bu(t) \\ y(t) \ &= \ C_x(t) + Du(t) \\ A &= \begin{bmatrix} 0 & 1 & 0 & \ldots & 0 \\ 0 & 0 & 1 & \ldots & 0 \\ \vdots & \vdots & & \ddots & \vdots \\ 0 & 0 & 0 & \ldots & 1 \\ -a_0 & -a_1 & -a_2 & \ldots & -a_{n-1} \end{bmatrix}, B = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} \\ C &= [b_0 \ b_1 \ b_2 \ldots b_{n-1}], \ D = 0 \end{aligned}$$

The linear system of 1st-order ODEs is called the state-space realisation of the nth-order ODE. Starting from the initial condition $x(0) = x_0$, the continuous-time linear system $x' = Ax + Bu$ has the unique solution $x(t)$ obtained employing the Lagrange's formula:

$$x(t) \ = \ e^{At}x_0 \ + \int e^{A(t-\tau)}Bu(\tau)d\tau \tag{A.6}$$

where the first term is the natural response of the system (i.e., if no external input is applied), while the second term is the forced response. Given $x(0)$ and $u(t)$, $\forall$ t $\in$ [0, T], Lagrange's formula allows computing $x(t)$ and $y(t)$, $\forall$ t $\in$ [0, T].

Let us introduce now different classes of dynamical systems.

A system is said Linear time-varying (LTV) if A, B, C, and D are constant matrices.

$$\begin{cases} \dot{x}(t) = A(t)x(t) + B(t)u(t) \\ y(t) = C(t)x(t) + D(t)u(t) \end{cases}$$

A generalisation of LTV systems is linear parameter-varying (LPV) systems:

$$\begin{cases} \dot{x}(t) = A(p(t))x(t) + B(p(t))u(t) \\ y(t) = C(p(t))x(t) + D(p(t))u(t) \end{cases}$$

A system is multivariable if it can have m inputs ($u(t) \in \mathbb{R}^m$) and p outputs ($y(t) \in \mathbb{R}^p$).

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t) \\ y(t) = Cx(t) + Du(t) \end{cases}$$

For linear systems, it still holds that $A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times m}$, $C \in \mathbb{R}^{p \times n}$ *and* $D \in \mathbb{R}^{p \times m}$.

In nonlinear systems, the evolution of the system and the output are characterised via nonlinear functions.

More precisely, f: $\mathbb{R}^{n+m} \rightarrow \mathbb{R}^n$, g: $\mathbb{R}^{n+m} \rightarrow \mathbb{R}^p$ are (arbitrary) nonlinear functions.

$$\begin{cases} \dot{x}(t) = f(x(t),\ u(t)) \\ y(t) = g(x(t),\ u(t)) \end{cases}$$

Time-varying nonlinear systems:

$$\begin{cases} \dot{x}(t) = f(t, x\ (t),\ u(t)) \\ y(t) = g(t, x(t),\ u(t)) \end{cases}$$

A.3.1.1 Stability

Let us consider the following continuous-time nonlinear system:

$$\begin{cases} \dot{x}(t) = f(x(t),\ u(t)) \\ y(t) = g(x(t),\ u(t)) \end{cases}$$

Definition: A state $x_r \in \mathbb{R}^n$ and an input $u_r \in \mathbb{R}^m$ are an equilibrium pair if for initial condition $x(0) = x_r$ and constant input $u(t)\ \ u_r$ the state remains constant: $x(t) = x_r$, $\forall$ $t \geq 0$.

x_r is called the equilibrium state, while u_r equilibrium input.

Now let us consider the nonlinear system:

$$\begin{cases} \dot{x}(t) = f(x\ (t),\ u_r) \\ y(t) = g(x(t),\ u_r) \end{cases}$$

Being x_r an equilibrium state, $f(x_r, u_r) = 0$.

Definition: The equilibrium state x_r is stable if $\forall\ \epsilon > 0\ \ \delta > 0$: $\|x(0)\text{-}x_r\| < \delta \Rightarrow \|x(0)\text{-}x_r\| < \epsilon$, $\forall\ t \geq 0$.

In other words, if for each initial condition $x(0)$ 'close enough' to x_r, the corresponding trajectory $x(t)$ remains near x_r, $\forall\ t \geq 0$.

The equilibrium point x_r is called asymptotically stable if it is stable and $x(t) \longrightarrow x_r$ for $t \longrightarrow$. Otherwise, the equilibrium point x_r is called unstable.

A.3.1.2 Linearisation of nonlinear systems

Consider the nonlinear system:

$$\begin{cases} \dot{x}(t) = f(x(t),\ u(t)) \\ y(t) = g(x(t),\ u(t)) \end{cases}$$

Let (x_r, u_r) be an equilibrium, $f(x_r, u_r) = 0$. The linearisation aims to investigate the dynamic behaviour of the system under small perturbations $\Delta u(t) \triangleq u(t) - u_r$ and $\Delta x(0) \triangleq x(0) - x_r$. The evolution of $\Delta x(t) \triangleq x(t) - x_r$ is given by

$$\begin{aligned} \dot{\Delta x}(t) &= \dot{x}(t) - \dot{x}_r = f(x(t), u(t)) \\ &= f(\Delta x(t) + x_r, \Delta u(t) + u_r) \\ &\approx \underbrace{\frac{\partial f}{\partial x}(x_r, u_r)}_{A} \Delta x(t) + \underbrace{\frac{\partial f}{\partial u}(x_r, u_r)}_{B} \Delta u(t) \end{aligned}$$

$$\Delta y(t) \approx \underbrace{\frac{\partial g}{\partial x}(x_r, u_r)}_{C} \Delta x(t) + \underbrace{\frac{\partial g}{\partial u}(x_r, u_r)}_{D} \Delta u(t)$$

Similarly, where $\Delta y(t) \triangleq y(t) - g(x_r, u_r)$ is the perturbation of the output from its equilibrium.

The perturbations $\Delta x(t)$, $\Delta y(t)$, and $\Delta u(t)$ are (approximately) ruled by the linearised system:

$$\begin{cases} \dot{\Delta x}(t) = A\Delta x(t) + B\Delta u(t) \\ \Delta y(t) = A\Delta x(t) + D\Delta u(t) \end{cases}$$

A.3.1.3 Discrete-time systems

Discrete-time models describe relationships between sampled variables $x(kT_s)$, $u(kT_s)$, $y(kT_s)$, $k = 0, 1, \ldots$. The value $u(kT_s)$ is kept constant during the sampling interval $[kT_s, (k+1)T_s)$. A discrete-time signal can either represent the sampling of a continuous-time signal or be an intrinsically discrete signal.

Let us consider the set of n first-order linear difference equations forced by the input $u(k) \in \mathbb{R}$.

$$\begin{cases} x_1(k+1) = a_{11}x_1(k) + \ldots + a_{1n}x_n(k) + b_1u(k) \\ x_2(k+1) = a_{21}x_1(k) + \ldots + a_{2n}x_n(k) + b_2u(k) \\ \quad\vdots \qquad\qquad \vdots \qquad\qquad \vdots \\ x_n(k+1) = a_{n1}x_1(k) + \ldots + a_{n\,n}x_n(k) \quad + b_nu(k) \\ \quad x_1(0) = x_{10}, \qquad \cdots \qquad x_n(0) = x_{n0} \end{cases}$$

The system can be rewritten using a compact matrix form:

$$\begin{cases} x(k+1) = Ax(k) + Bu(k) \\ x(0) = x_0 \end{cases}$$

where x = [x1, …, xn] ∈ ℝ.

The solution is

$$x(k) = \underbrace{A^k x_0}_{natural\ response} + \underbrace{\sum_{i=0}^{k-1} A^i Bu(k-1-i)}_{forced\ response}$$

where the first term is the natural response, and the second term is the force response.

Let us consider the following discrete-time nonlinear system:

$$\begin{cases} x(k+1) = f(x(k),\ u(k)) \\ \quad y(k) = f(x(k),\ u(k)) \end{cases}$$

Definition: A state $x_r \in \mathbb{R}n$ and an input $u_r \in \mathbb{R}m$ are an equilibrium pair if for initial condition $x(0) = x_r$ and constant input $u(k)\ \ u_r$, $\forall\ k \in \mathbb{N}$, the state remains constant: $x(k)\ \ x_r$, $\forall\ k \in \mathbb{N}$.

x_r is called equilibrium state, u_r equilibrium input.

Definition: The equilibrium state x_r is stable if $\forall\ \epsilon > 0\ \ \delta > 0$: $\|x(0)\text{-}x_r\| < \delta \Rightarrow \|x(k)\text{-}x_r\| < \epsilon$, $\forall\ k \in \mathbb{N}$.

In other words, if for each initial condition $x(0)$ 'close enough' to x_r, the corresponding trajectory $x(k)$ remains close to x_r, $\forall\ k \in \mathbb{N}$.

The equilibrium point x_r is called asymptotically stable if it is stable and $x(t) \longrightarrow x_r$ for $t \longrightarrow$. Otherwise, the equilibrium point x_r is called unstable.

A.4 Control strategy design

A control strategy is an algorithm that takes decisions regarding the control actions to apply at each instant in time, i.e., it calculates the input values. Accordingly, the control strategy design involves developing a set of systematic approaches and methodologies to regulate the behaviour of a process.

The design task of a control strategy is an iterative process that may involve revisiting and refining the strategy based on performance evaluations and changes in the system or requirements. It requires a combination of theoretical knowledge, practical experience, and a thorough understanding of the specific process being controlled. For instance, a potential approach involves endeavouring to replicate, or model, the conduct of a human operator, whether real or hypothetical, following the expert system methodology. An alternative option is to strive for comprehension, or modelling, of the process behaviour, subsequently applying systematic methods within the realm of automatic control. These methods aim to formulate an appropriate control strategy.

Some common steps and considerations in control strategy design are the following:

- System analysis: analysis and understanding of the behaviour of the process; identification of key variables, parameters, and interactions that could affect the process.
- Definition of objectives: define the goals and objectives of the control system, so as to have a precise understanding of the desired outcomes.
- Modelling: a control strategy can be formulated based on a mathematical model, or it may explicitly incorporate a mathematical model as a means of dynamically assessing the effectiveness of specific control actions in real time. Hence, it is crucial to develop a mathematical model that represents the dynamics of the process, as accurately as possible.
- Controller selection: identification of the most appropriate control strategy and controller type based on the characteristics and objectives of the process.
- Tuning: fine-tune the controller parameters to achieve the desired process response. The task may involve simulation, analysis of stability criteria, and iterative adjustments to optimise the controller's performance.
- Robustness/Sensitivity analysis: understand how the performance of the process is affected by variations in parameters or external disturbances.
- Perform sensitivity analysis to understand how variations in parameters or disturbances impact the system's response.
- Testing and validation: test and validate the control strategy under various operating conditions and scenarios. This step ensures that the control strategy performs as intended and meets the specified objectives.

A.4.1 Open-loop and closed-loop control

Open loop control is a type of control system where the control action is applied to the process independently of the process output. In other words, the controller does not use feedback from the output to determine its actions but operates only based on the initial input or predetermined settings. The control action is based solely on the input or predefined settings.

Let us assume a process model of the type $y = f(u,d)$, being y, u, d the outputs, the inputs, and the disturbances vectors, while f is a generalised operator. Moreover, assume the availability of desired output values y_d. If the relation is invertible, it is possible to obtain $u_d = f^{1}(d,y_d)$, which corresponds to an open-loop control strategy as depicted in Figure A.4.

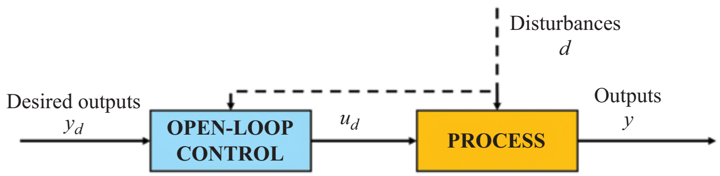

Figure A.4 Schematic representation of an open-loop control approach

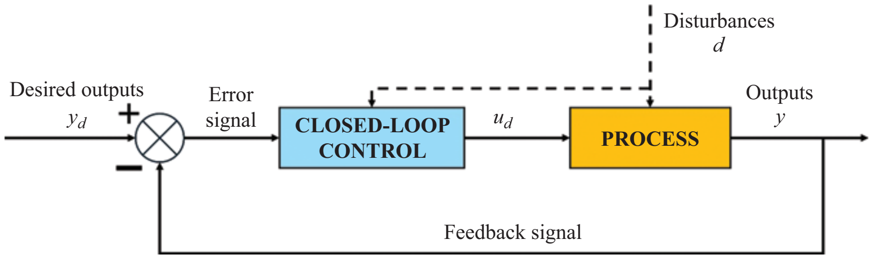

Figure A.5 Schematic representation of an closed-loop control approach

This control approach is generally simple to design, and if the process itself is stable then there is no risk of obtaining an unstable system. For these reasons, such a control scheme is suitable for situations where the relationship between input and output is well-known and does not change. However, the approach has some major disadvantages:

- Accuracy: since there is no feedback, open loop systems are not so accurate and can be affected by disturbances or changes in the system parameters;
- Inefficiency: May overdo or underdo the task;
- No error correction: Cannot correct any deviations from the desired output.

Because of these drawbacks, any automatic control system needs to include a closed-loop structure.

Closed loop control is a type of control system where the control action is dependent on the output of the system. The system uses feedback to compare the actual output with the desired output and make adjustments to minimise the error. This feedback loop allows the system to be more accurate and responsive to changes and disturbances. The control scheme can be represented in Figure A.5. The main characteristics of closed-loop control are:

- Feedback mechanism: the system continuously monitors the output and compares it to the desired setpoint. Any deviation from the setpoint is corrected by adjusting the input.
- Self-correcting: closed-loop systems can automatically correct errors by compensating for disturbances and changes in the system dynamics.
- Not simple design: require sensors, controllers, and actuators to implement the feedback mechanism.
- High accuracy: the feedback mechanism ensures accuracy and reliability

A.5 Traffic control approaches

A.5.1 Traffic control based on continuous models

Traffic control approaches based on continuous models leverage advanced mathematical and computational techniques to optimise traffic flow in real time by treating traffic as a fluid-like entity. These models, such as the Lighthill–Whitham–Richards (LWR) model, describe traffic dynamics through partial differential equations, capturing variations in traffic density, flow rate, and speed. Continuous traffic control strategies, including adaptive traffic signal control, ramp metering, and variable speed limits, rely on real-time data from sensors and predictive algorithms to dynamically adjust control measures. This real-time optimisation helps mitigate congestion, reduce travel times, and improve overall traffic efficiency. The related traffic control schemes rely on a robust infrastructure of sensors and data processing capabilities, enabling a real-time, proactive approach to traffic management that significantly reduces congestion and improves overall traffic efficiency. Despite the complexity and high data requirements, continuous models offer high scalability and predictive capabilities, making them crucial for modern, intelligent traffic management systems.

As stated in Siri *et al*. (2021), traffic control schemes falling into this category can be further divided into in-domain and boundary control. The first category refers to control inputs that are continuous both in time and space over the whole road stretch or network; the latter, instead, refers to control inputs acting at the boundaries of the considered road area.

A.5.1.1 Feedback control

Continuous time feedback control is a control strategy where the system continuously monitors its output in real time and adjusts its input based on the feedback to achieve the desired performance. The main idea is to minimise the difference between the actual output and the desired setpoint by using a feedback loop that operates continuously over time. This approach enhances the efficiency, safety, and reliability of transportation systems by dynamically responding to traffic conditions.

Most control laws depend on measurements of the state of a portion of the system and integrate the Lighthill–Whitham–Richards (LWR) model. For instance, let us consider a first-order model of the following form:

$$\partial Q(x,t)/\partial t + \partial/\partial x(u(x,t)f\ (Q(x,t))) = 0 \tag{A.7}$$

where $u(x,t)$ is the control input.

If one employs a variable speed limit on a road stretch as a control action, then the model becomes:

$$\partial Q(x,t)/\ \partial t + \partial/\partial x\ (F(Q(x,t)l(x,t))) = 0 \tag{A.8}$$

Being $l(x,t) \in (0,\ 1]$ the ratio between the imposed speed limit and the maximum speed; $F(Q,l)$ is a nonlinear function defining the Fundamental Diagram in which the variable speed limit is integrated.

A.5.1.2 Backstepping control

Backstepping control is an advanced control strategy used to deal with nonlinear systems. It systematically constructs a stabilizing controller by breaking down a complex nonlinear system into smaller subsystems and designing control laws for each step, ensuring stability at every stage. In traffic engineering, backstepping control can be applied to various scenarios, such as adaptive traffic signal control, ramp metering, and vehicle platooning.

An example of backstepping control is in Yu and Krstic (2018), where a downstream ramp metering (named DORM) and an upstream ramp metering (named UORM) control strategies were proposed. The DORM strategy consists of a boundary feedback law, while the UORM consists of a full-state feedback boundary control. The control law is obtained via a backstepping transformation.

A.5.1.3 Optimal PDE control

Optimal control of Partial Differential Equations (PDEs) involves the use of mathematical techniques to derive control strategies that optimise the performance of traffic systems modelled by PDEs. These models typically describe the evolution of traffic density, flow, and speed over time and space. The goal is to minimise congestion, travel time, or other performance criteria while ensuring smooth and safe traffic flow. The key components are: the macroscopic traffic flow model, objective function, and control input.

Then, to formulate an optimal control problem, the following steps are required:

- Identification of the state equation. This allows us to define the main variables of the macroscopic traffic flow model (e.g., LWR model) employed to emulate the behaviour of a road stretch or network. It is important to correctly define the state variables,
- Definition of control constraints, i.e., the physical and operational constraints on the control inputs and input variables (e.g., umin $\leq$ u(x,t) $\leq$ umax),
- Optimisation problem. This step requires the definition of the objective function to minimise (or to maximise) subject to the state equation and constraints.

As an example, Delle Monache *et al*. (2017) proposed an optimal controller for flow regulation via a variable speed limit. The considered single road is represented with the LWR model, and the optimal problem aims to minimise the quadratic difference between the achieved outflow and the given target flow, considering that the state of the system only depends on free-flow conditions.

A.5.2 Traffic control based on discrete models

Discrete-time control strategies involve making decisions at specific time intervals based on traffic data collected at those intervals. These strategies can be applied to various aspects of traffic management, such as signal control, ramp metering, and dynamic speed limits. The control decisions are updated periodically rather than continuously, which can simplify implementation and reduce computational requirements.

The key common components of discrete-time control strategies are:

- Data collection: Traffic data is collected at regular intervals (e.g., every minute, every 5 min).
- Control intervals: The system updates control actions at discrete time steps (e.g., $t = 0$, T, 2T, …)

- Control laws: specific algorithms that determine control actions based on the current state of the system and desired objectives.

A.5.2.1 Feedback control

Discrete-time feedback control involves updating traffic control strategies at specific time intervals based on the real-time feedback of the traffic state. This approach is particularly useful for adaptive traffic signal control, ramp metering, and variable speed limits, where timely adjustments can significantly enhance traffic flow and reduce congestion.

A general formulation for this type of controller is presented as follows:

$$g_{k+1} = g_k + K_p(e_k) + K_i \sum_{i=0}^{k} e_i + K_d(e_k - e_{k-1})$$

where k is the discrete time step, e_k is the error at time step k, K_p K_i, and K_d are proportional, integral, and derivative gains, respectively. The error e_k is the difference between the actual value of a state variable and the desired one. Most common controllers employ solely proportional or proportional-integral actions.

This kind of controller can be extended to take into account the state variables measured in a larger portion of the road network. This is useful since in some cases the control action does not depend only on the traffic state of a portion of the road network but on a wider area.

A.5.2.2 Optimal control

Optimal control methods aim to optimise a certain performance criterion, taking into account the evolution of the dynamical system over a given time horizon. The choice of inputs u is usually limited due to physical or technical constraints that define an admissible control region via a set of inequalities This leads to optimal (or suboptimal) solutions according to a predefined objective function.

A general formulation for optimal controller is presented in the following:

Given the initial conditions $x(0) = x0$ of the system and the estimated sequence of disturbance inputs $d(k)$, $k = 0, \ldots, K$-1, find the optimal control sequence $u(k)$, $k = 0, \ldots, K$-1 that minimises the criterion J.

$$J = \vartheta[\underline{x}(K)] + \sum_{k=0}^{K-1} \varphi[\underline{x}(k),\ \underline{u}(k),\ \underline{d}(k)]$$

subject to the system dynamics and constraints (e.g., control variable constraints $u_{\min} \le u(k) \le u_{\max}$, $k = 0, \ldots, K$-1.

The solution to the problem is the optimal state trajectory $x^{\circ}(k)$, $k = 1, \ldots, K$ and the optimal control inputs $u^{\circ}(k)$, $k = 0, \ldots, K$-1 over the considered time horizon.

In the case of urban traffic, this criterion typically corresponds to the minimisation of total travel time in the considered network.

A.5.2.3 Model predictive control

Model Predictive Control (MPC) approaches allow to control a dynamic system in real-time on the basis of the prediction of its evolution over a finite horizon, considering an objective function to optimise. This is achieved by iteratively solving a Finite-Horizon Optimal Control Problem that is updated on the basis of real system measurements and can account for different types of constraints.

Considering a generic MPC scheme in a discrete-time framework, the problem to be solved at each time step k, $k = 0, \ldots, K$, over a given prediction horizon of K_p time steps can be stated as:

Given the initial conditions on the system state $x(k)$ and the estimated sequence of exogenous inputs $d(h)$, $h = k, \ldots, k+K_p$-1, find the optimal control sequence $u(h)$, $h = k, \ldots, k+K_p$ -1 that minimizes:

$$J(k) = \vartheta[\underline{x}(k + K_p)] + \sum_{h=k}^{k+k_{p-1}} \varphi[\underline{x}(h),\ \underline{u}(h), \underline{d}(h)]$$

subject to

$$x(h + 1) = \underline{f}[\underline{x}(h),\ \underline{u}(h),\ \underline{d}(h)]\ h = k, \ldots, k + K_p - 1$$

$$u^{\min} \leq \underline{u}(h) \leq u^{\max}\ h = k, \ldots, k + k_p - 1$$

And further possible additional constraints on the state variables $x(h)$, $h = k+1, \ldots, k+K_p$, and on the control variables $u(h)$, $h = k+1, \ldots, k+K_p$.

The solution to the problem is the optimal state trajectory $x^\circ(h|k)$, $h = k+1, \ldots, k+K_p$ and the optimal control input sequence $u^\circ(h|k)$, $h = k, \ldots, k + K_p$-1 over the considered prediction horizon.

According to the receding-horizon framework, only the first z elements (with $z < h$) of the optimal control sequence are actually implemented in the system.

A.6 Summary

This appendix explores the core elements of automation control, with a special focus on its use in traffic management. As the complexity of contemporary transportation networks increases, the incorporation of automation control is becoming essential for improving efficiency, safety, and sustainability.

This section aims to equip readers with a comprehensive understanding of automation control principles, specifically adapted to address the unique requirements and challenges of traffic systems.

References

Some suggestions for further readings are reported below among the huge literature available on these topics.

Delle Monache, M. L., Piccoli, B., and Rossi, F. (2017). Traffic regulation via controlled speed limit. *SIAM Journal on Control and Optimization*, 55(5), 2936–2958.

Papageorgiou, M. (1998). Automatic control methods in traffic and transportation. In *Operations Research and Decision Aid Methodologies in Traffic and Transportation Management* (pp. 46–83). Berlin, Heidelberg: Springer.

Siri, S., Pasquale, C., Sacone, S., and Ferrara, A. (2021). Freeway traffic control: A survey. *Automatica*, 130, 109655.

Yu, H., and Krstic, M. (2018). Traffic congestion control on Aw-Rascle-Zhang model: Full-state feedback. In *2018 Annual American Control Conference (ACC)* (pp. 943–948). IEEE.