Automation of Energy Systems

Alberto Leva

Department of Electronics, Information and Bioengineering
Politecnico di Milano, Italy

Academic year 2022/2023

Introduction

Course *rationale* and overview

A few words on history

Some bare essential definitions

A first glance at control problems

Recap, needs and next steps

Foreword

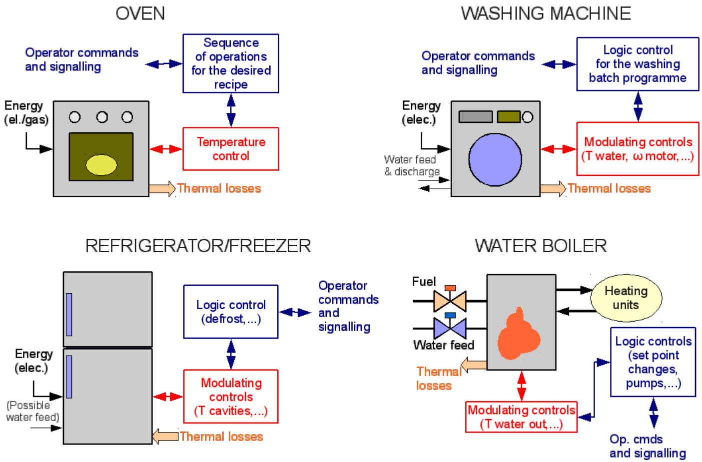
- Systems that produce, distribute and use energy are becoming more and more complex and articulated:
  - different sources (renewable or not);
  - different types of generation (e.g., centralised vs. distributed);
  - complex markets in rapid evolution;
  - ...
- Therefore, energy system experience an increasing need for automation
  - at more and more levels (from the power plant to the town grid, down to the single house);
  - more and more integrated (e.g., to coordinate the generation and use of electricity and heat);
  - and for new demands (comfort, economy, environmental impact,...).

Foreword

- Purpose of the course:
  - address the *scenario* just sketched
    - providing the student with a *system-level* view – typical of the Automation Engineer – on the encountered control problems, the solutions adopted for them to date, and the possible future developments
    - avoiding details on the various types of generators, utilisers and so forth—a matter to which specialised courses are devoted.
- *Caveat emptor*:
  - an exhaustive treatise of the matter is absolutely impossible, even at quite high and abstracted a level;
  - thus we shall proceed by introducing general concepts and then going through illustrative examples abd cases;
  - it will be the duty of *you engineers* to abstract, the lessons learnt, generalise and transpose the underlying concepts where applicable.

Where is automation in energy systems?

Let us analyse some introductory schemes



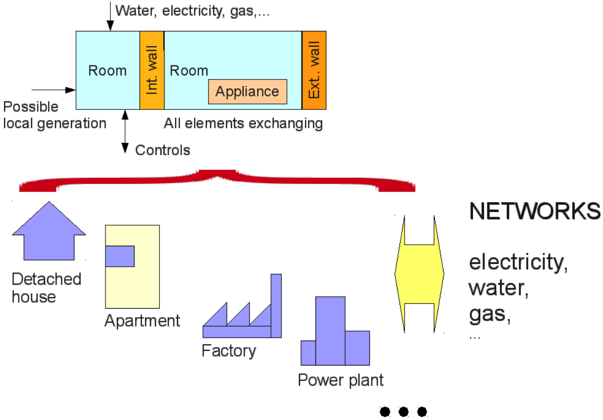
Where is automation in energy systems?

A bit of generalisation

- Carrying on with the list:
  - generators (fossil, hydro, solar, wind, geothermal,...);
  - other "utilisers" (heating elements, fan coils, AHUs,...);
  - maybe some home/building automation, not installed (only) for energy purposes... ...but surely with a relevant energy effect;
  - ...
- Scaling up:
  - "larger" components (building or compound-level HVAC,...);
  - industrial machines/installations;
  - large power plants;
  - ...

Where is automation in energy systems?

Let us now start *aggregating*



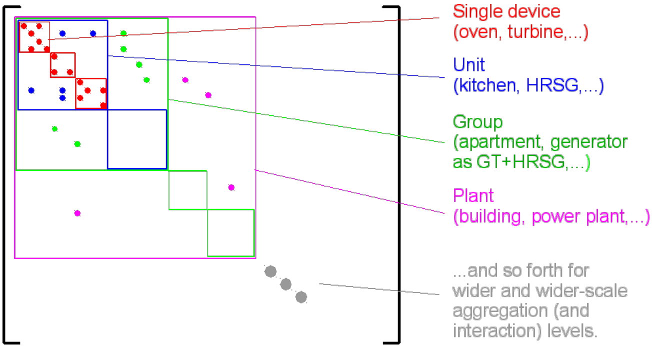
NOTE: each part of the system has its own controls, and must somehow coordinate with the others.

Summing up...

- Each object (or aggregate of objects) is generally controlled for *local* goals:
    - cook some food as quickly as possible;
    - maximise the economic revenue of a household photovoltaic generator based on forecasts of weather, energy use and prices of electricity and gas;
    - generate the total power required at any time by the national network while maintaining each plant as close as possible to its optimal operating point and without overloading the transmission lines;...)
- bearing however in mind that any action on that individual object will (more or less) influence the overall system.

Is there any *hierarchy* in energy systems?

- Rigorously, no.
- To understand, let us fictitiously imagine to write the whole dynamic model of the "world" energy system and observe its *incidence matrix* (a boolean matrix showing which variables appear in which equations):



However, structuring is needed

- Apparently, the problem needs *decentralised* solutions — at far smaller scale than the planet ☺
- Therefore, some hierarchy - or better, as we shall see, some *structuring* of the problems is in order.
- In other words, we need
  - to understand which are the relevant problems (a task already carried out, although the matter is continuously evolving, think e.g. of environmental issues)
  - and that can reasonably be dominated (which is often not trivial),
  - find for them solution that are sound from an engineering standpoint
  - and figure out how to put said solutions at work also when the boundary conditions for the problem at hand vary, which can also be due to how the *rest* of the system is affected by the introduced solutions.

Consequences

- What do we mean "a problem can be dominated"?
- Essentially that once said problem is stated in system-theoretical terms
  - having extended its size (i.e., the set of described phenomena) enough for the "rest of the world" to be properly represented by boundary conditions and/or disturbances (i.e., exogenous entities)
  - it is possible to find for it a solution of acceptable complexity and implementable with information available in practice.
- An important concept in this regard is that of "dynamic separation", on which we shall return in due time.

Consequences

- To state and address problems this way, we need a "systemic" approach, in which components
    - can be described at different detail levels,
    - but preserving their interfaces with other components of the overall system,
    - and as independently as possible of how they are connected to the rest of the system;
- at the same time, the approach should allow to state problems in such a way to be tractable with well established control methodologies and techniques (although the energy context has been fostering new ones).
- In addition, simulation techniques play a crucial role.

Coverage

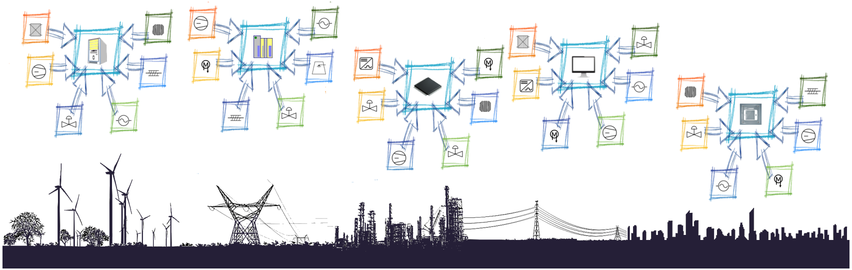
— or, “where is automation” revisited in a more strict sense

Controlled "system of systems":
producers, utilisers, storage, transport networks...

Coverage

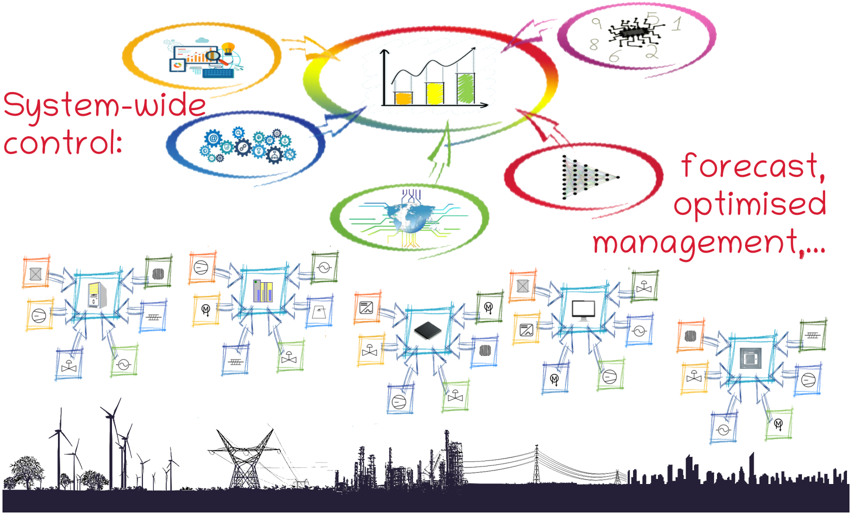
— or, “where is automation” revisited in a more strict sense

Modulating loops & logic at "peripheral" level: unit management, response to local events,...



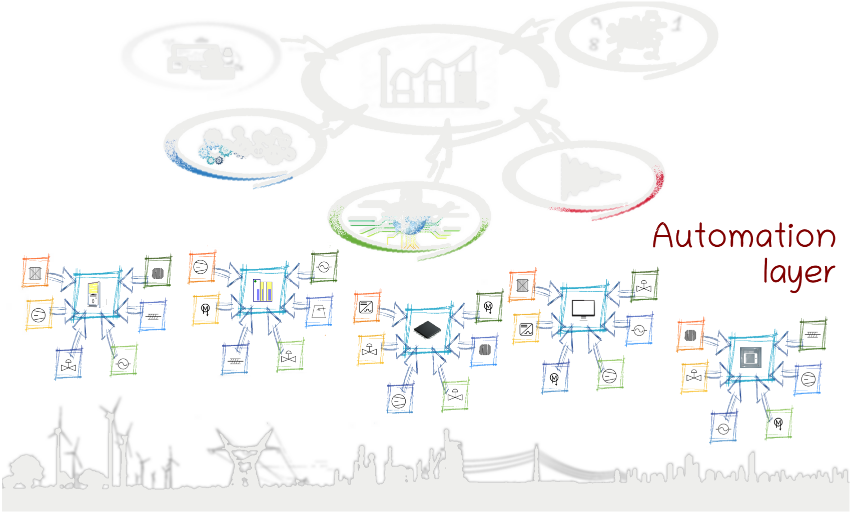
Coverage

— or, "where is automation" revisited in a more strict sense



Coverage

— or, "where is automation" revisited in a more strict sense



An automation layer as a contributor to "smartness"

— i.e., an important aspect of our focus

- **Wikipedia**
  Smart systems incorporate functions of sensing, actuation, and control in order to describe and analyse a situation, and make decisions based on the available data in a predictive or adaptive manner, thereby performing smart actions. In most cases the "smartness" of the system can be attributed to autonomous operation based on closed loop control, energy efficiency, and networking capabilities.

- Make thus local- and mid-level controls efficient, and as a consequence
  - local events will be managed locally instead of scaling up to involve system-wide controls,
  - so that the said controla will see a "simpler" managed object
  - to the advantage of operation optimality.

A few words on history

- How has automation in energy systems developed?
- Initially (and trivially) to be able to run them: without control systems one can
  - neither operate a power plant
  - nor avoid the electric network collapse
  - nor maintain the required pressures in a gas network extending for tens of thousands of kilometers;
- then (and by the way almost immediately) for efficiency reasons concerning "big" system components (e.g., power plants),
- but always having in mind, more or less explicitly, a fairly well defined structure of the system (in the electric case, for example, a few large generators, transmission, distribution, and many utilisers of variable size but small if compared to the generators).

A few words on history

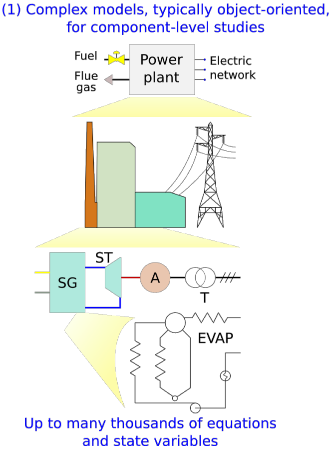
- Today, however, the *scenario* has changed:
  - there is more and more distributed generation to be integrated with the network,
  - different energy sources, both renewable and not, are being considered,
  - economies of scale are being exploited (think e.g. of district heating),
  - integration is spreading out among controls (e.g., feedforward compensation from heat-releasing appliances to room temperature controllers)...
  - and sometimes among "machines" (e.g., heat recovery, generation surplus management, and so forth).

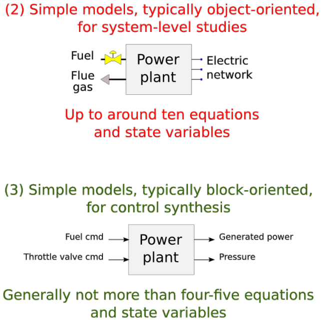
Summing up...

- Energy systems provide a number of control problems to address, and several are "new".
- Many solutions that were acceptable in the past are no longer acceptable today (efficiency demands are becoming more stringent).
- In the course are we thinking to address everything we have mentioned so far?
- Not at all: it would take too long and would not even be a "smart" idea.
- Let us therefore review the initial statement about the purpose of the course as we can now give it a more precise meaning...

Summing up...

...by means of another scheme





- (1) and (2) have the same interfaces;
- (1–3) must be consistent;
- in this course we concentrate on (2–3) and only sketch out relationships with (1).

Course organisation

General information

- The course consists of
  - about 30 hours of lectures
  - and about 20 hours of classroom practice, including some computer laboratory activity,

  for a total of 5 CFU.
- Lectures are guided with slides,
- while classroom practice/laboratory involves guidance as well as individual work.

Course organisation

Synopsis of lecture and practice subjects

- Introduction (this lecture);
- review of the mathematical, modelling and control principles that will be used later on;
- the main physical objects (generators, utilisers, distribution networks) involved in energy systems:
  - synthetic description and dynamic behaviour in the context where they operate,
  - simple models, parametrisable with the minimum necessary information;
- the main control problems in energy systems and the strategies to handle them.

Classroom practice sessions, interlaced with lectures, contain examples and case studies concerning small applications, and involve the use of open source simulation and control synthesis tools.

Instructor information, online resources, contacts

- Alberto Leva
  Dipartimento di Elettronica, Informazione e Bioingegneria
  Politecnico di Milano
  Voice (39) 02 2399 3410
  E-mail `alberto.leva@polimi.it`

- Resources:
  - course page on WeBeep (`https://webeep.polimi.it`, course ID NNNNNN),
  - git repository at
    `https://github.com/looms-polimi/Automation_of_Energy_Systems`,
  - old web page at `https://leva.faculty.polimi.it` → Teaching.

- Office hours:
  - at the DEIB (2nd floor, room 234) on Thursday, 10.30 to 12.30
    (if coming on purpose drop a line the day before)
  - or by appointment.

Teaching material

- Course slides (*in fieri*);
- some literature and web references, that will be introduced later on.

Software

All free (as free speech, not free beer)

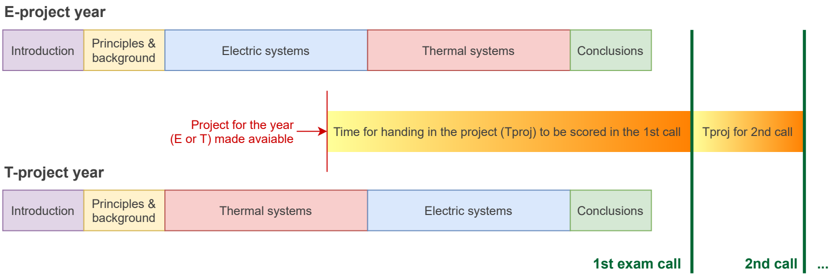
- **Scilab**:
  analysis, synthesis and simulation of causal dynamic systems (block-oriented approach).
  `https://www.scilab.org`
- **OpenModelica**:
  modelling and simulation allowing for a-causal components (object-oriented approach).
  `https://www.openmodelica.org`
- **wxMaxima**:
  CAS (Computer Algebra System) for symbolic computations.
  `https://wxmaxima-developers.github.io/wxmaxima`
- **Important note**: the goal of this course is not to teach (let alone compare) approaches to modelling and simulation, nor is it to train students for the use of one or another software; the applications above are *only* tools for putting the concepts learnt to work.

Exam

- The exam is made of a project and a written test.
- The project is scored 18–30L (33) and contributes 40% of the total score.
- The remaining 60% comes from the written test (approx. 1h30").
- The written test is scored up to 30L (33) as well.
- To pass the exam, the written test score must be 16/30 or above; lower scores cannot be compensated for by the project.
- You can hand in the project only once; no exceptions.
- If you re-enroll to the course for another year, the project is still valid.
- On the other hand you can repeat the test as many times as you wish (*cum grano salis*, please).
- But starting – not handing in, beware – a written test, erases any previous test score.

Exam

- You must hand in the project before taking the written test.
- To make this feasible, thanks to some modularity in the course, the timeline is as follows (box widths are not to scale with time):



- E-project and T-project years alternate. This year is E-project.
- The written test is always on the entire course.
- The project theme (electric or thermal) changes every year — yes the project score is valid forever anyway, to prevent useless questions ☺

Exam

- For the project, you have to form teams of four.
- You will solve an energy-related automation problem, of reasonable size and complexity, using the methods and tools taught in the course.
- You will then create a slide-based screencast (maximum 15 minutes, sharp) to show your results; the presentation structure is totally free and up to the team.
- Only two constraints (for a motivation, search "pitch presentation rules"):
  1. all members of the team must talk, no matter order and distribution;
  2. at 15 minutes from the start I stop the player, whatever comes after does not exist.
- You will finally pack the screencast, the slides and the required code (details in due course) into a single compressed file, and deliver this via BeeP.
- Clarity and presentation quality will be taken into account in the evaluation process — I do not mind a lot about *video* quality as long as I can understand, though.
- In the event of significant doubts, I might ask for a colloquium with a team (arranged on a per-case basis).

Exam

Privacy note — pedantic but required, apologies

- We are managing project groups by means of a shared worksheet on OneDrive.
- Access will be possible only by authenticating with the polimi online services.
- This means sharing your name, registration No. and `@polimi` e-mail with the class.
- No project evaluation is ever shared (outside the team).

- If you object to the above, please send me an e-mail: silence for one week starting today will be taken as “all right, I am fine with this”.
- Objecting persons – hopefully few, but feel free – will be managed separately (as said intra-team info sharing is inevitable, however).
- And now, let us begin the course.

Energy sources classification

- **Primary – Secondary Energy (PE–SE)**
  - Primary energy is that gathered directly in nature:
    coal, oil, natural gas, biomass, radioactive substances, geothermal energy, wind, sun radiation, gravitational potential energy (e.g. for hydroelectric generation),...
  - Secondary energy is obtained by transforming primary energy in a form easier to use/store/transmit:
    electricity, fossil fuels, hydrogen, steam...
- **Renewable – Non Renewable Energy Sources (RES–NRES)**
  - Renewable energy is obtained from (practically) inexhaustible sources (and without pollutant release):
    sun, wind, tides, geothermal heat,...
  - Non renewable energy implies consuming some "fuel" (and releasing some pollutant):
    fossil fuels,...

$\Rightarrow$ Extracting, transforming and transmitting energy are industrial processes where automation is required.

Energy intensity, conservation, efficiency

- **Energy intensity**
  Amount of energy per unit of intended (i.e., useful) result.
  - EI of a country: energy consumption per GDP unit;
  - EI of a product: (average) energy consumption per product unit;
  - EI of a service: (average) energy consumption per served request;
  - ...
- **Energy conservation**
  Reducing the (growth of) energy consumption in absolute physical terms.
- **Energy efficiency**
  Reducing the EI of a product/service/whatever while preserving the intended result (e.g., less energy for HVAC with the same comfort).

$\Rightarrow$ Conservation and efficiency are separate but intertwined concepts, and automation (to say nothing of process/control co-design) is useful to pursue them.

Some control problems – fundamental

- **Generator control (PE→SE)**
  - Minimise fuel consumption (NRES) – maximise caption (RES),
  - that is, stay in the vicinity of "optimal" operating points.
- **Utiliser control (SE→final use)**
  - Maintain functional quality (room temperatures, correct appliance operation,...).
- **Transmission control (one type of SE)**
  - Maintain generation–demand balance;
  - Manage storages if applicable;
  - Maintain network operation quality (voltage and frequency, gas pressures,...);
  - Minimise network losses;
  - Avoid network overloading.

Some control problems – advanced

- **Generator energy mix control (NRES)**
  - Mix PEs (e.g., oil and gas)
  - to fulfil SE demands (e.g., electricity and steam)
  - optimising for cost, pollutant emission, or any combination thereof.
- **Utiliser energy mix control**
  - Mix available SEs (e.g., electricity, gas and solar heat stored as hot water)
  - to fulfil final use needs (e.g., electric and thermal loads of a house)
  - optimising for any Key Performance Indicator (KPI) related to the utiliser process.
- **Zone- (or neighbourhood-)level energy mix control**
  - Given a zone with various generators (both RES and NRES) and utilisers,
  - find the optimal management of each generator and utiliser
  - and also of storages where applicable
  - optimising for...? Defining zone-level KPIs is an open problem and often involves conflicts among stakeholders' interests (e.g., we all cook dinner in the evening but gas suppliers prefer flat demand profiles).

Control problems – main common characteristics

- Set point tracking and/or functional minimisation;
- set point profiles generation;
- scheduling (of generation, utilisation,...);
- disturbance rejection;
- uncertainty (e.g., demand forecast errors, time-varying costs,...);
- hard constraints (e.g., generator operational limits,...);
- soft constraints (e.g., regulations/agreements on acceptable transient overloads, reserve management,...);
- robustness versus plant–model mismatches (e.g., line impedance variations owing to weather conditions,...);
- fault tolerance (e.g., in the case of generator breakdowns).

Recap

- Automation in energy system means controlling generation, transmission, storage and utilisation...
- ...in a coordinated manner
- and for multiple objectives at different system levels.

- Control problems call for a coordinated use of various techniques
- and solutions need implementing in heterogeneous architectures.

- Process/control co-design is helpful wherever applicable
- but refurbishing of already designed systems is of paramount relevance.

Needs

- A modelling framework;
- quantitative performance indicators (automation-oriented KPIs);
- a problem taxonomy;
- best practices and control design guidelines based on the above,
- and corresponding validation/assessment methodologies;
- awareness of implementation-related facts.

Next steps

- Review modelling principles;
- define a modelling framework suitably managing components' behaviours and interfaces in both the object-oriented (OO) and the block-oriented (BO) context, conceptually relating the two;
- devise the required component models (of course we shall not exhaust this matter) and the major KPIs;
- learn to create system-level compound models tailored to handling problems that involve those KPIs.

Part A

Principles and background

Modelling principles

Modelling and control

First-principle *versus* data-based modelling

Balance equations

An explanatory example

Lessons learnt

Preliminary definitions

- Modulating control:
  - the controller outputs are real numbers (within given intervals);
  - the natural formalism is continuous- or discrete-time dynamic systems.
- Logic control:
  - the controller outputs are lexical variables (on/off, forward/stop/backward...);
  - the natural formalism is discrete-events dynamic systems.

Modelling and – or better, *for* – control (in energy systems)

A look at the encountered problems

- Control problems are classically divided into process and motion ones.
- We shall now
  - first discuss the system-theoretical side of the above division,
  - then point out that the so introduced viewpoint on a control problem has consequences
    - on the characteristics that a model has to possess to be "good for control",
    - on the way a control scheme needs structuring,
    - and on the *rationale* to follow for tuning the blocks in that scheme.
  - and finally focus for this part of the course on models (the other items above shall be addressed later on).
- We shall also start introduce simple examples with an explanatory purpose.

Process *versus* motion control

Focus on modulating control — by abstraction from an example

- Process control — typical case, a chemical plant:
  - set points are seldom – and most often slowly – modified — think e.g. of the desired temperature profile on a distillation column;
  - the main role of control is to reject disturbances, and these can be of large and unpredictable entity, sometimes up to exceeding short-term actuator capability — e.g., weather on the distillation column above;
    - in the long run if their effect is "averaged" (e.g., the quality of a product flow is averaged by filling tanks),
    - or also in the short run if this counts (e.g., when even a brief over-temperature may generate spurious species and compromise a reaction);
  - hence, multiple time scales are almost ubiquitous ans span a wide band — e.g., seconds or less for flows, hours or days for temperatures;
  - somehow correspondingly, the control system tends to show a clear hierarchy and schemes composed of quite numerous blocks — e.g., a plant-wide MPC could optimise set points for plant units, where in turn local MPCs determine local set points for further low-level loops or structures, mostly based on simple blocks like PIDs.

Process *versus* motion control

Focus on modulating control — by abstraction from an example

- Motion control — typical case, a machining center:
  - set points are modified continuously and in general rapidly
    — think e.g. of the desired trajectory and speed of a tool;
  - the main role of control is to track set points, and these are quite often known *a priori*
    — e.g., many operations are repetitive;
  - disturbances are present, though in general quite well known/predictable, but controllers and actuators must recover as fast as required
    — e.g., variable payloads or material characteristics;
  - there can be multiple time scales, but not much widespread (say ms to s) and in general not so dependent on the particular application
    — e.g., not as heterogeneous as reactor residence times;
  - the control system hierarchy, as well as the individual schemes, is in general quite simple (not exactly the same for *logic* control, but details on this would stray from our scope).

Process *versus* motion control

Focus on logic control

- Process control:
  - the system maintains an operating regime thanks to modulating control;
  - logic control takes care of startup, shutdown, emergencies, and little else.
- Motion control:
  - the system carries out working sequences governed by logic control;
  - modulating control is mostly confined within machines to take care of positions, speeds and the like.
- Observing that of course the scheme we are discussing is a crude simplification, let us now move the focus to the context of energy systems.

Focus on energy systems

and on models as a consequence of problems

- In energy systems, "process-type" control problems tend to dominate...
- ...but nonetheless, from the abstracted system-theoretical viewpoint, those systems offer a mix of "process-type" and "motion-type" control problems,
- and most relevant, the balance is changing
  - as the traditional situation of a few large generators *versus* many small loads (we shall return on this) is becoming less general,
  - and as renewables are moving in with increasing importance;
- problems (thus models) are multi-physics
  — most typically hydraulic, thermal, and electrical (mechanics appears essentially in rotating masses);
- distributed-parameter systems, hence Partial Differential Equations (PDEs) appear, but generally in one spatial coordinate
  — e.g., for piping.

Preliminary definitions

- First-principle models:
  - rely on physical laws and/or empirical correlations;
  - natively continuous-time, possibly switching (the latter fact not of particular interest for us in this course);
  - discrete-time version of course possible;
  - parameters have a direct physical interpretation;
  - can describe objects that do not yet exist;
  - can be built with components validated individually prior to being assembled together;
  - the interfaces of the said components being suitable for attributing a physical meaning as well (pins, flanges,...);
  - may have to be complex if physics call for this.

Preliminary definitions

- Data-based models:
  - rely on reproducing measured data;
  - hence of course cannot model the not yet existing;
  - inherently discrete-time as so is data;
  - structure either "partially suggested" by the phenomenon to describe (grey box)
  - or just chosen as the best to fit the data (black box);
  - parameters have in general no physical meaning,
  - hard to make modular unless orientation or causality (what is input and what is output) is decided *a priori*,
  - which is not always possible if the same component needs connecting with others in different or even time-varying manners;
  - may be able of "summarising" complex physics with simple structures (on a per-case basis, however).

Foreword

- We need
  - mass, energy and momentum equations for thermo-hydraulic networks (for space reasons we only treat incompressible fluids with a single species);
  - energy equations for solid bodies such as walls;
  - (semi)empirical equations for flow/pressure relationships and heat transfer;
  - equations for electric networks (phasor-based and reduced to a single phase since this simplification does not hinder explaining the required concepts);
  - energy equations for rotating masses (e.g., in alternator-based electric generators);
- Of course we deal with *dynamic* balance equations;
- we now proceed to a quick illustration by (main) application area, then discuss an example, and finally abstract the lessons learnt.

Thermo-hydraulic networks

Mass equation

- The equation is written with reference to a *control volume*.
- Let M be the (incompressible, single species fluid) mass contained in the volume.
- Let w_i, $i = 1 \dots n_m$, be the n_m mass flowrates exchanged by that volume with the external environment, considered positive if entering the volume.
- The equation then simply reads

$$\frac{dM(t)}{dt} = \sum_{i=1}^{n_m} w_i(t)$$

where t is obviously the (continuous) time.

Thermo-hydraulic networks

Energy equation

- Also this equation is written with reference to a control volume containing a total energy E and where n_m mass flowrates w_i and n_h heat rates Q_j enter (if positive).
- The time derivative of E is the sum of the heat rates Q_j, not associated to any mass transfer, and of the energy contribution yielded by the mass flowrates.
- The latter contributions are of two *coexisting* types:
  - (signed) heat transfers inherent to mass transfers, taking the form

$$\text{mass flowrate} \times \text{fluid specific energy } ([kg/s] \times [J/kg] = [J/s] = [W])$$

  - and work exerted by the entering fluid on that contained in the volume or *vice versa*, that in differential and specific form reads

$$d\mathscr{L} = d(pv) = d(p/\rho)$$

    where v is specific volume, ρ is density, and $d\mathscr{L} = pdv + vdp$; pdv is "compressing work" and vdp "impelling work" (both signed).

Thermo-hydraulic networks

Energy equation

- Thus, the thermodynamic variable characterising the energy contribution of an entering/exiting mass flowrate to a volume, is the fluid specific enthalpy

$$h = e + \frac{p}{\rho}$$

  where e is the specific internal energy.
- The energy equation then takes the form

$$\frac{dE(t)}{dt} = \sum_{i=1}^{n_m} w_i(t)h_i(t) + \sum_{j=1}^{n_h} Q_j(t).$$

- For incompressible fluids, at pressures and temperatures of interest for us, e largely dominates p/ρ; hence, *in these conditions* we can approximately take $h = e = cT$, where c is the fluid specific heat (assumed here constant) and T its temperature.

Thermo-hydraulic networks

Energy equation

- Furthermore here we only deal with single-species fluids, thereby having always to do with a single specific heat c.
- Given all the above, for our purposes the energy equation for fluids is

$$c\frac{dM(t)T(t)}{dt} = c\sum_{i=1}^{n_m} w_i(t)T_i(t) + \sum_{j=1}^{n_h} Q_j(t).$$

where T is the control volume temperature, assumed uniform (hence adopting a finite-volume approach to distributed-parameter systems, though we do not further discuss this matter) and M the fluid mass.

Thermo-hydraulic networks

Energy equation

- Sometimes the mass is constant, like in a tube section always filled, whence

$$cM\frac{dT(t)}{dt}=c\sum_{i=1}^{n_m}w_i(t)T_i(t)+\sum_{j=1}^{n_h}Q_j(t).$$

- In other cases, such as tanks, this is not true. It is then convenient to expand the derivative on the left hand side, and subtract the mass equation multiplied by cT; this provides

$$\begin{array}{cccccccc}
cM(t)\frac{dT(t)}{dt} & + & cT(t)\frac{dM(t)}{dt} & = & & c\sum_{i=1}^{n_m}w_i(t)T_i(t) & + & \sum_{j=1}^{n_h}Q_j(t) \\
 & - & cT(t)\frac{dM(t)}{dt} & = & - & cT(t)\sum_{i=1}^{n_m}w_i(t) & & \\
\hline
cM(t)\frac{dT(t)}{dt} & & & = & & c\sum_{i=1}^{n_m}w_i(t)(T_i(t)-T(t)) & + & \sum_{j=1}^{n_h}Q_j(t)
\end{array}$$

that is sometimes called the "net energy" equation.

Thermo-hydraulic networks

Momentum equation

- This equation is used primarily for modelling tubes (valves are a somehow analogous case briefly treated later on).
- Consider a tube (element) and write that the time derivative of the fluid momentum is the sum of the forces acting on it, that is,
  - pressure forces at the two ends,
  - gravity force,
  - and friction force on the lateral surface,

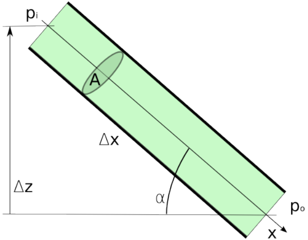
  all projected onto the tube *abscissa* x
- In fact, other components do not act on the fluid motion on the *prevailing dimension* x and merely result in constraint reaction forces, not relevant for the energy-related aspects on which we focus.

Thermo-hydraulic networks

Momentum equation

- To keep complexity at a level compatible with the course, consider a tube with uniform section A (and recall that we only deal with incompressible fluids).
- This yields

$$M\frac{du(t)}{dt} = Ap_i(t) - Ap_o(t) + Mg\sin(\alpha) - f_a(t).$$



- Note that $\sin(\alpha) = \Delta z/\Delta x$, Δz being the initial altitude minus the final one
- while the friction force f_a, always contrasting motion, is

$$f_a = K_f A_\ell \rho u|u|$$

where A_ℓ is the lateral surface.

Thermo-hydraulic networks

Momentum equation

- The quantity K_f is called *friction coefficient*, depends on the fluid/wall contact characteristics, and is tabulated for most cases of interest based on experiments and empirical correlations.
- The equation obviously contains an inertia term, that in our models can however be omitted.
- This is possible because hydraulic phenomena are much faster than thermal ones, which are our main subject.
- In other words, since thermal variables (such as temperatures) propagate at the fluid speed while hydraulic ones (such as pressures and flowrates) propagate at the speed of sound in the fluid, we can safely assume that for our purposes "hydraulics is always at steady state".
- Thus, we can write the momentum equation as the algebraic one

$$A\left(p_i(t) - p_o(t)\right) + Mg\frac{\Delta z}{\Delta x} - K_f A_\ell \rho u(t)|u(t)| = 0.$$

Thermo-hydraulic networks

Momentum equation

- Summing up, denoting by A the tube (uniform) section, by L its length, by ω its internal perimeter, simplifying the notation a bit and recalling that $w = \rho A u$, we have

$$A(p_i - p_o) + \rho A L g \frac{\Delta z}{L} - K_f \omega L \rho u|u| = 0$$

$$p_i - p_o = K_f \frac{\omega L}{\rho A^3} w|w| - \rho g \Delta z.$$

- In addition, if the tube is installed in such a way that w has always the same sign, taken positive when going from the higher- to the lower-pressure end, we can write

$$p_i - p_o = K_f \frac{\omega L}{\rho A^3} w^2 - \rho g \Delta z,$$

that we shall often further summarise as

$$p_i - p_o = \frac{K_T}{\rho} w^2 - \rho g \Delta z.$$

Thermo-hydraulic networks

Momentum equation

- The momentum equation can also represent the typical valve behaviour.
- This means neglecting more than one phenomenon, that however is of interest only for sizing, or operating conditions not advised for "good" plant management—i.e., not of interest here.
- From our point of view, consider a valve like a variable-section short tube installed so that the flow does not reverse. Thus, take

$$p_i - p_o = \frac{K_T}{\rho} w^2 - \rho g \Delta z,$$

set $\Delta z = 0$ (short component, hardly any gravity effect) and rewrite as

$$w = C_v \Phi(x) \sqrt{\rho (p_i - p_o)}$$

where C_v is the *flow coefficient*, $x \in [0,1]$ the command, and $\Phi(x), \Phi(0) = 0, \Phi(1) = 1$ the *opening* or *intrinsic characteristic*.

Solid bodies

Energy equation

- In this case we naturally neglect any volume effect, and consider the specific heat spatially uniform (non-homogeneous walls will be treated with at least one equation per material layer).
- Since there is no mass transfer, the equation simply reads

$$cM\frac{dT(t)}{dt} = \sum_{j=1}^{n_h} Q_j(t).$$

where symbols have the same meaning as in the fluid case, and M is of course constant.

Heat transfer equations

Foreword

- These are algebraic equations, as they describe no storage.
- We need to model
  - conduction within solids (and sometimes fluids),
  - convections between a solid and a fluid,
  - and radiation.
- In all cases we shall adopt simplified concentrated-parameter descriptions right from the beginning.

Heat transfer equations

Conduction

- We shall only use a simplified planar descriptions, as more detailed ones would stray from our scope.
- The heat rate from the a to the b side of a solid layer is

$$Q_{ab} = G(T_a(t) - T_b(t))$$

  where T_a and T_b are the side temperatures.
- The thermal conductance G is

$$G = \lambda \frac{A}{s}$$

  where λ is the material's thermal conductivity, A the layer surface, and s its thickness.

Heat transfer equations

Convection

- The convective heat rate from a solid wall (subscript w) to a fluid (subscript f) is

$$Q_{wf} = \gamma A (T_w(t) - T_f(t))$$

  where T_w and T_f are the wall and a fluid "bulk" temperature, while A is the contact surface.
- The thermal exchange coefficient γ can be considered constant (as we shall almost always do) or made dependent on the fluid and motion conditions, typically with relationships involving the Reynolds (for forced convection) or Grashof (for natural one), Nusselt and Prandtl numbers.

Heat transfer equations

Convection

- A common, somehow intermediate refinement is to have γ just depend on the fluid velocity tangent to the wall.
- To this end, taking a reference heat exchange coefficient value γ_0 as corresponding to a reference velocity u_0, a widely used relationship is

$$\gamma(t) = \gamma_0 \left(\frac{u(t)}{u_0} \right)^{0.8}$$

where $u(t)$ is the fluid velocity.

Heat transfer equations

Radiation

- At a simplified level, the radiative heat transfer from a body a to a body b depends on the difference of their absolute (Kelvin) temperatures to the fourth power, i.e.,

$$Q_{ab} = K\left(T_a^4 - T_b^4\right).$$

- The radiative heat transfer coefficient K depends on several things, including the bodies' emissivity and their view factors.
- However in this course the only relevant case will be that of solar radiation, that can be very naturally viewed as a prescribed power flux $[W/m^2]$.

Electric networks

Foreword

- In this course we shall deal essentially with AC power networks.
- The matter is vast, and simplifications are introduced so as to transmit the necessary concept with the minimum complexity sufficient to explain them.
- In detail (and somehow anticipating) we shall assume
  - a single-frequency synchronous network (quite reasonable if frequency is well controlled, and we do not have the time to deal with the connected "stability" problems),
  - all generators described by a constant voltage behind their internal reactance,
  - linear behaviour of transmission lines,
  - no transformers (we only spend some words on reactive power control) as doing so significantly reduces computations,
  - a one-phase (or equivalently, a *balanced* multiphase) system.
- We shall thus adopt a *phasor-based* modelling approach.

Electric networks

Phasors

- Any quantity varying (co)sinusoidally with constant frequency ω can be represented as
$$A\cos(\omega t+\theta) = \Re\left(Ae^{j(\omega t+\theta)}\right) = \Re\left(Ae^{j\theta}\,e^{j\omega t}\right)$$
where j is the imaginary unit, $e^{j\omega t}$ yields time dependence, and the phasor $Ae^{j\theta}$ magnitude and phase with respect to a convenient reference.
- This allows for a phasor arithmetic to handle AC networks with frequency "hidden"...
- ...in the term $e^{j\omega t}$ and in the value of frequency-dependent impedances.
- Recall – for the last time – that phasor analysis is for synchronous networks with constant frequency (whence its alternative name as "static analysis").

Electric networks

Basic equations (essentially to agree notation...)

- Ohm's law $\underline{V} = \underline{Z}\underline{I}$ or $\underline{I} = \underline{Y}\underline{V}$, where $\underline{V}, \underline{I}$ are voltage and current (phasors) and $\underline{Z}, \underline{Y}$ the complex impedance and admittance, respectively (underline indicates complex numbers). We shall typically express $\underline{Z}$ as $R + jX$ $R, X \geq 0$, and $\underline{Y}$ as $G - jB$ ($G \geq 0$, and mind the minus to have $B \geq 0$), where R, X, G, B are respectively called resistance $[\Omega]$, reactance $[\Omega]$, conductance $[S]$, and susceptance $[S]$.
- Kirchoff's laws (nothing to say here).
- Power (* denotes the complex conjugate):

$$
\begin{array}{lll}
\text{complex} & \underline{S} = \underline{V}_{RMS}\underline{I}^*_{RMS} = P + jQ = Ae^{j\phi} & \\
\text{apparent} & A = |\underline{S}| = V_{RMS}I_{RMS} = V_{max}I_{max}/2 & [VA], \\
\text{active} & P = \Re(\underline{S}) = V_{RMS}I_{RMS}\cos\phi & [W], \\
\text{reactive} & Q = \Im(\underline{S}) = V_{RMS}I_{RMS}\sin\phi & [VAR], \\
 & \cos\phi & \text{power factor.}
\end{array}
$$

For a sinusoidal signal F, $F_{RMS} = F_{max}/\sqrt{2}$.

Energy equation for rotating masses

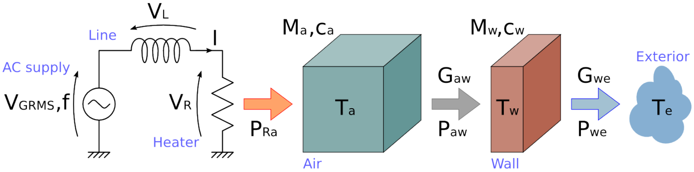
- Many electric generators contain rotating masses, like turbine and alternator rotors.
- Their angular velocity affects the generated frequency (to be controlled).
- The energy equation states that the time derivative of the kinetic energy equals the algebraic sum of powers, i.e.,

$$\frac{d}{dt}\left(\frac{1}{2}J\omega_r^2\right) = P_m - P_e$$

where J is the inertia, ω_r the angular velocity, P_m the mechanical power applied to the shaft (positive if entering the considered machine) and P_e the *active* electric power (positive if generated).

Our first multi-physics model

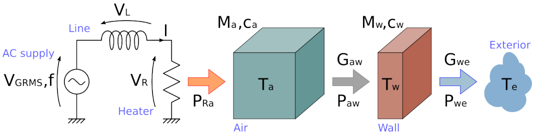
- Crude simplification of an electrically heated room:



- Data:
  - room dimensions $4m \times 4m \times 3m$ height;
  - only one side wall (30cm thick) exchanges heat, the others are adiabatic;
  - no openings, external temperature $5°C$;
  - no heater losses (all of its power is released to air);
  - air density $1.1 kg/m^3$, specific heat $1020 J/kg°C$;
  - wall density $2000 kg/m^3$, specific heat $800 J/kg°C$;
  - air–wall heat exchange coefficient $10 W/m^2°C$;
  - wall–exterior heat exchange coefficient $4 W/m^2°C$;
  - AC supply voltage $220 V\,RMS$, $f = 50 Hz$, $R = 50 \Omega$, $L = 10 mH$.

First model (M1)

with the electric part described in the time domain



$V_G(t) = V_{GRMS}\sqrt{2}\sin(2\pi f t)$	Source generator
$V_R(t) = RI(t)$	Heater resistor
$V_L(t) = LdI(t)/dt$	Line inductor
$V_R(t) + V_L(t) = V_G(t)$	KVL
$P(t) = V_R(t)I(t)$	Active power
$Q(t) = V_L(t)I(t)$	Reactive power (addendum)
$P_{Ra}(t) = P(t)$	All *active* power to air
$M_a c_a dT_a(t)/dt = P_{Ra}(t) - P_{aw}(t)$	Air energy balance
$M_w c_w dT_w(t)/dt = P_{aw}(t) - P_{we}(t)$	Wall energy balance
$P_{aw}(t) = G_{aw}A_w(T_a(t) - T_w(t))$	Air–wall heat transfer
$P_{we}(t) = G_{we}A_w(T_w(t) - T_e(t))$	Wall–exterior heat transfer
$T_e(t) =$	Exogenous temperature

Note the partition into electric and thermal equations; each has its boundary conditions, and one such condition is presented by the former to the latter set.

Second model (M2)

with the electric part described in the phasor domain

- First let us compute P with Maxima by issuing the commands

```
IRMS : VGRMS/(R+%i*w*L);
S    : VGRMS*conjugate(IRMS);
P    : realpart(S);
```

- This yields

$$P = \frac{RV_{GRMS}^2}{R^2 + (\omega L)^2},$$

where obviously $\omega = 2\pi f$, hence M2 is

$P(t) = RV_{GRMS}^2/(R^2 + (\omega L)^2)$	Active power
$M_a c_a \dot{T}_a(t) = P_{Ra}(t) - P_{aw}(t)$	Air energy balance
$M_w c_w \dot{T}_w(t) = P_{aw}(t) - P_{we}(t)$	Wall energy balance
$P_{Ra}(t) = P(t)$	All *active* power to air
$P_{aw}(t) = G_{aw} A_w (T_a(t) - T_w(t))$	Air–wall heat transfer
$P_{we}(t) = G_{we} A_w (T_w(t) - T_e(t))$	Wall–exterior heat transfer
$T_e(t)$	Exogenous temperature

Note: from now on we shall often use a dot to indicate time derivatives.

Time scales for the thermal part

- Compute the transfer function $G(s)$ from P_{Ra} to T_a:

```
Paw      : Gaw*(Ta-Tw);
Pwe      : Gwe*(Tw-Te);
se1      : Ma*ca*Tadot = PRa-Paw;
se2      : Mw*cw*Twdot = Paw-Pwe;
solxdot  : solve([se1,se2],[Tadot,Twdot])[1];
solTadot : rhs(solxdot[1]);
solTwdot : rhs(solxdot[2]);
A        : jacobian([solTadot,solTwdot],[Ta,Tw]);
B        : jacobian([solTadot,solTwdot],[PRa,Te]);
Tmat     : ratsimp(invert(s*ident(2)-A).B);
G        : Tmat[1,1];
```

- Result:

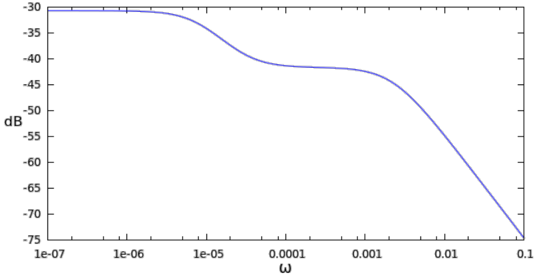
$$G(s) := \frac{T_a(s)}{P_{Ra}(s)} = \frac{c_w M_w s + G_{we} + G_{aw}}{c_a c_w M_a M_w s^2 + (c_w G_{aw} M_w + c_a(G_{we} + G_{aw}) M_a)s + G_{aw} G_{we}}$$

Time scales for the thermal part

- Put numbers in and plot the Bode magnitude diagram:

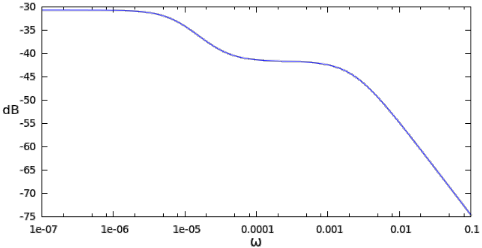
```
load("bode");
Gnum(s)  := ev(subst([Ma=4*4*3*1.1,ca=1020,Gaw=4*3*10,
                      Mw=4*3*0.3*2000,cw=800,Gwe=4*3*4],G));
bode_gain(Gnum(s),[w,1e-7,1e-1]);
```

- Result:



Time scales for the thermal part

- Interpretation:



Two time scales:

- one ("fast") relative to the energy storage in the air – order of magnitude, $1000\,s$,
- the other ("slow") relative to wall storage – order of magnitude, some $10^5\,s$, i.e., some days.

- Hence, electric phenomena (time scale dominated by $L/R = 0.2\,ms$) are about 10^5 times faster than the faster thermal phenomena, which are in turn about 10^4 times faster than the slower thermal ones.

Modelica implementation of M1 and M2

```
model M1
  parameter Real w=6.28*50;
  parameter Real R=50;
  parameter Real L=0.01;
  parameter Real VGRMS=220;
  Real VG,VL,VR;
  Real I(start=0),P,Q;
  parameter Real Ma=4*4*3*1.1;
  parameter Real ca=1020;
  parameter Real Gaw=4*3*10;
  parameter Real Mw=4*3*0.3*2000;
  parameter Real cw=800;
  parameter Real Gwe=4*3*4;
  Real PRa,Paw,Pwe,Te;
  Real Ta(start=10);
  Real Tw(start=10);
equation
  VG           = VGRMS*sqrt(2)*sin(w*time);
  VG           = VL+VR;
  VR           = R*I;
  VL           = L*der(I);
  P            = VR*I;
  Q            = VL*I;
  Ma*ca*der(Ta) = PRa-Paw;
  Mw*cw*der(Tw) = Paw-Pwe;
  PRa          = P;
  Paw          = Gaw*(Ta-Tw);
  Pwe          = Gwe*(Tw-Te);
  Te           = 5;
end M1;
```

```
model M2
  parameter Real w=6.28*50;
  parameter Real R=50;
  parameter Real L=0.01;
  parameter Real VGRMS=220;
  parameter Real Ma=4*4*3*1.1;
  parameter Real ca=1020;
  parameter Real Gaw=4*3*10;
  parameter Real Mw=4*3*0.3*2000;
  parameter Real cw=800;
  parameter Real Gwe=4*3*4;
  Real P,PRa,Paw,Pwe,Te;
  Real Ta(start=10);
  Real Tw(start=10);
equation
  P            = R*VGRMS^2/(R^2+w^2*L^2);
  Ma*ca*der(Ta) = PRa-Paw;
  Mw*cw*der(Tw) = Paw-Pwe;
  PRa          = P;
  Paw          = Gaw*(Ta-Tw);
  Pwe          = Gwe*(Tw-Te);
  Te           = 5;
end M2;
```

Simulation of M1 and M2

for comparison (OpenModelica 1.17.0 on Ubuntu 18.04, i7-3520M CPU @2.90GHz, 16GB RAM)

- CPU time for simulating two hours (DASSL variable-step solver):

Model M1	11.79	s
Model M2	3.95	ms

- Average errors on air temperature and heating power:

$$
\begin{aligned}
\bar{e}_{Ta} &= \frac{1}{7200}\int_0^{7200}\left(T_{a,M1}(t)-T_{a,M2}(t)\right)dt &= 2.5\cdot 10^{-4}\,{}^\circ C \\
\bar{e}_{P_{Ra}} &= \frac{1}{7200}\int_0^{7200}\left(P_{Ra,M1}(t)-P_{Ra,M2}(t)\right)dt &= 2.8\cdot 10^{-2}\,W
\end{aligned}
$$

- With an average power of approximately 1 kW, *from the thermal viewpoint* the two models are practically equivalent; however, thanks to not representing the electric time scale in detail, model M2 simulates almost 3000 times faster than M1.

Model analysis for control

- The relationship between heater command (control signal) and heater power is very fast, hence from the thermal viewpoint algebraic.
- The transfer function from heater command to air temperature (controlled variable) can be written as

$$G(s) = \mu \frac{1+s\tau_z}{(1+s\tau_1)(1+s\tau_2)}.$$

- Let us see a less extreme example of spread dynamics than the room case, namely $\mu = 1$, $\tau_1 = 100$, $\tau_2 = 1$, $\tau_z = 10$.

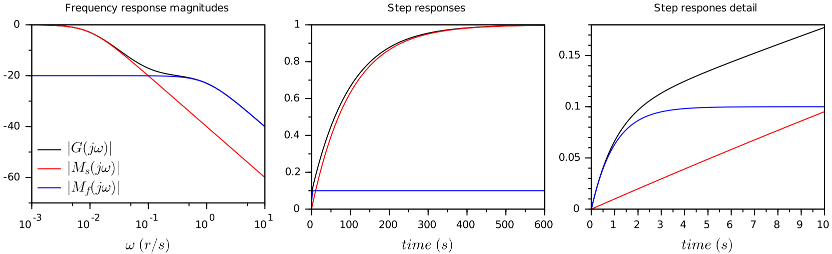
- Question: which is the “best” 1st order approximation in this case for setting up a controller?

Model analysis for control

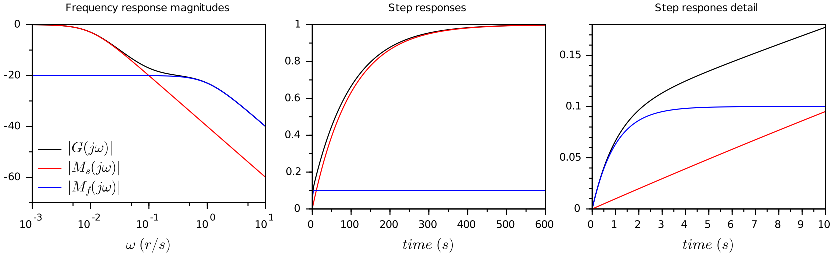
- Answer: it depends on which dynamics you want to control (in the room case, air or walls).
- Consider two 1st order models – $M_s(s)$ and $M_f(s)$ to name them – focusing on the slow and the fast dynamics respectively in $G(s)$, i.e.,

$$M_s(s) = \frac{1}{1+100s}, \quad M_f(s) = \frac{0.1}{1+s}.$$

- Observe the Bode diagrams and the step responses of $G(s)$, $M_s(s)$ and $M_f(s)$:

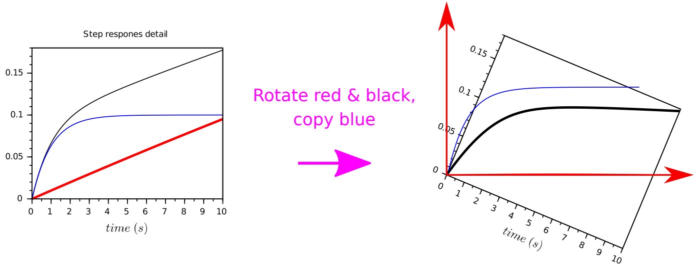


Model analysis for control



- Interpretation: a "fast" state variable moving relatively to a "slow" one.
- Model $M_f(s)$ is apparently unfit to reproduce the step response of $G(s)$...
- ...but what if you want a controller for a critical frequency in the vicinity of the fast dynamics?

Model analysis for control



- Further interpretation: at the required time scale I have to govern how T_a moves w.r.t. T_w — at that scale, T_w is not moving.
- In this respect, what does the figure above INTUITIVELY suggest?
- Exercise: set up a controller for $G(s)$ aiming at a closed-loop settling time around 10s, using the two approximated models, and comment.
- We shall come back to this matter when discussing tuning policies.

Lessons learnt

- Models must represent the phenomena of interest, and be efficient.
- Models must catch the control-relevant dynamics and time scales.
- Consequences:
  - one has invariantly to use sets of models (and data), or said otherwise, Model and Data Bases;
  - one must keep such knowledge bases consistent — research topic here, volunteers? 🤓
  - no control engineering culture without modelling culture.

Model structuring and tuning brush-up

Modelling framework

A bit of practice

Brush-up (and some complements) on controller tuning

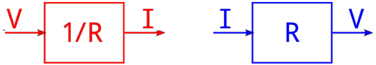
Block-oriented and object-oriented (BO/OO) models

- As said we stick to 1st-principle models (based on dynamic balances) since we may need to analyse/simulate/optimise something that does not yet exist (thus, no data to identify e.g. black- or grey-box models).
- This does *not* mean that identification and estimation never come into play – think e.g. of adaptive control – but rather just that the matter does not fit in this course (except for a few words later on).
- We need however to distinguish between block- and object-oriented models.
  - Block-oriented models
    - are oriented or causal,
    - thus written having their boundary conditions in mind,
    - and connect to one another via inputs and outputs.
  - Object-oriented models
    - are non oriented or a-causal,
    - thus written independently of their boundary conditions,
    - and connect to one another via ports.
- Let us go through an introductory example, and then generalise.

BO (causal) and OO (a-causal) models

Introductory example

- We need to model a resistor ⇒ Ohm's law.
- We consider two cases:
  1. the resistor is connected to a fixed voltage generator E, leading with obvious notation to the model
     $V = E$ Voltage generator (boundary condition for the resistor)
     $I = V/R$ Resistor
  2. the resistor is connected to a fixed current generator A, leading this time to
     $I = A$ Current generator (boundary condition for the resistor)
     $V = RI$ Resistor
- Same component, different boundary conditions, different models,
- both oriented: in the former case V is an input and I an output, in the latter *vice versa*; the two BO resistor models are



with *input* and *output* connectors.

BO (causal) and OO (a-causal) models

Introductory example – a perspective change

- Now we take a different approach:
  - we identify *ports*, i.e., physical terminals characterising the interface exposed by the modelled component to the outside. In this case ports are the resistor's two pins (a and b to name them) each of which carries a voltage V and a current I;
  - we write the component's *constitutive equations*, that with obvious notation (current is taken positive when entering the pin) read

$$RES : \begin{cases} a.I + b.I & = & 0 \\ a.V - b.V & = & Ra.I \end{cases}$$

  - we can do the same for the voltage and the current generators, obtaining (mind the current sign convention)

$$VGEN : \begin{cases} a.I + b.I & = & 0 \\ a.V - b.V & = & E \end{cases} \qquad CGEN : \begin{cases} a.I + b.I & = & 0 \\ a.I & = & -A \end{cases} .$$

  - we can finally introduce a ground (with a single pin a), i.e.,

$$GND : a.V = 0.$$

BO (causal) and OO (a-causal) models

Introductory example – exploiting OO modelling

- Doing so, the two addressed cases only differ for the generator equations:

Constitutive equations

$$\begin{aligned} RES.a.I + RES.b.I &= 0 \\ RES.a.V - RES.b.V &= RES.R * RES.a.I \end{aligned}$$ res

V gen:

$$\begin{aligned} GEN.a.I + GEN.b.I &= 0 \\ GEN.a.V - GEN.b.V &= GEN.E \end{aligned}$$

C gen:

$$\begin{aligned} GEN.a.I + GEN.b.I &= 0 \\ GEN.a.I &= -GEN.A \end{aligned}$$

$$GND.a.V = 0$$ Gnd

Connection equations

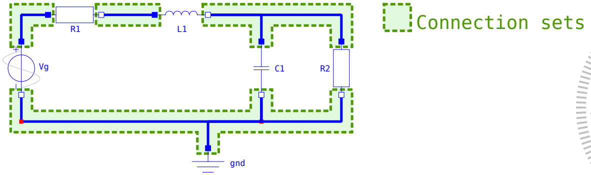
$$\begin{aligned} GEN.a.V &= RES.a.V \\ GEN.a.I + RES.a.I &= 0 \end{aligned}$$ pins GEN.a and RES.a

$$\begin{aligned} GEN.b.V &= GND.a.V \\ RES.b.V &= GND.a.V \\ GEN.b.I + RES.b.I + GND.a.I &= 0 \end{aligned}$$ pins GEN.b, RES.b and GND.a

BO (causal) and OO (a-causal) models

Introductory example – wrap-up

- Summing up and abstracting:
    - component models are written independently of their connections;
    - they are neither oriented nor closed (having less equations than variables);
    - they interconnect via *ports*;
    - and the overall model is closed (thereby determining orientation) by joining
        - the (component-specific) *constitutive equations*
        - and the (port-specific) *connection equations*,
        - which happens connecting ports to form *connection sets*, as exemplified below.



BO (causal) and OO (a-causal) models

Introductory example – wrap-up

- Ports (or connectors) carry variables, that can be
  - *effort* variables (defined as difference with respect to a reference, think of voltage or temperature),
  - or *flow* variables (defined as flowing through a boundary, think of current or power).
- Connecting N (two or more) ports into a *connection set* generates
  - $N-1$ equations per effort variable, stating that it is equal on all the connected ports,
  - and one equation per flow variable, stating that its sum over all connected ports is zero.
- Note: all of the above has a direct counterpart in OO modelling languages such as Modelica (more on this in due course).

BO and OO modelling

Distinctive features of both approaches

- Both types of models allow to *encapsulate* the model's behaviour with respect to its interface, hence permitting e.g. to scale the representation detail.
- BO models require the system to be oriented $\Rightarrow$ suitable for control components (block diagram elements) and *complete* controlled systems' models, i.e., for "plants completely built, with control signals and controlled variables already specified as inputs and outputs.
- OO models do not require the system to be oriented $\Rightarrow$ suitable for individual plant components.
- Quite intuitively, we shall use a combination of the two model types.

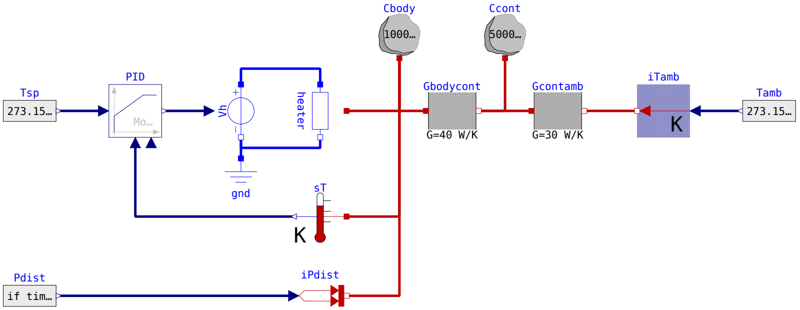
BO and OO modelling

Distinctive features of OO models

- In OO models, ports are naturally associated with energy transfer.
- Consider the typical port with one effort and one flow variable: most such couples are keen to be related to power.
- Examples:
  - electric, obviously (voltage v, current i) $\Rightarrow vi =$ power;
  - mechanic, traslational (position x, force f) $\Rightarrow \dot{x}f =$ power, note the derivative;
  - mechanic, rotational (angle φ, torque τ) $\Rightarrow \dot{\varphi}\tau =$ power, note again the derivative;
  - thermal, conductive and convective (temperature T, power Q): note that here power is *not* the product of the two.
- This OO characteristic allows to create electric equivalent models, particularly when the flow variable is linearly related to a difference in the effort one, and to make such models modularly composable.
- Electric equivalents are often useful for a rapid interpretation of an interconnected system (we shall see some examples).

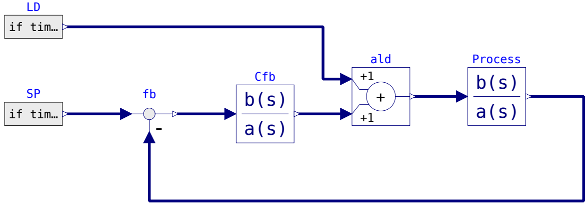
A multi-physics model

(electrical, thermal, and analogue control domains)



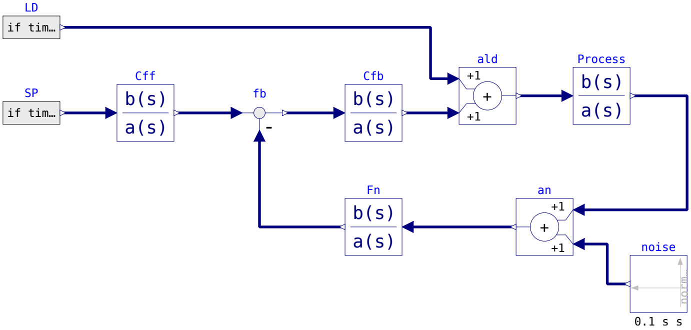
- Entirely assembled with components from the Modelica Standard Library (MSL for short).
- Observe the co-existence of BO and OO components.
- Check out `https://modelica.org/libraries`.

A basic control loop



- Components from the MSL, package `Blocks`.

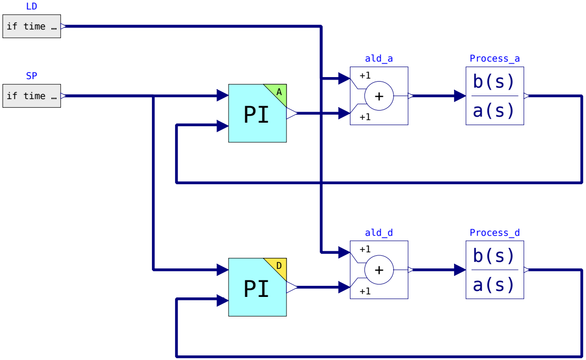
A more articulated control loop



- Two-degree-of-freedom (2-dof) controller (more on this later).
- Measurement noise and analogue filter in the feedback path.

A simple PI control loop

with analogue and digital controller representation



- Controllers from the course library.
- Play around with these models, create others, experiment, enjoy ☺

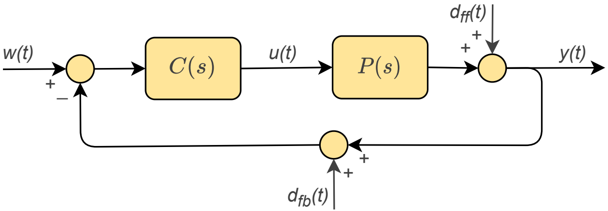
Foreword

Purpose of this treatise

- In the sequel of the course we shall need to tune several controllers.
- In most cases, these will be of PI/PID type (albeit as part of larger schemes).
- We need "approximate but fast" procedures to do that.
- We therefore brush up the tuning matter with a deliberate operational focus.
- We also take the occasion of introducing some concepts (servo vs. regulatory tuning, 2-dof exploitation. *a priori* robustness quantification) to be used later on in the addressed control schemes...
- ...and of general validity also beyond the energy domain.

Foreword

The addressed *scenario* (in the Bode applicability hypotheses)



- Loop $L(s)$, sensitivity $S(s)$, complementary sensitivity $T(s)$, control sensitivity $Q(s)$:

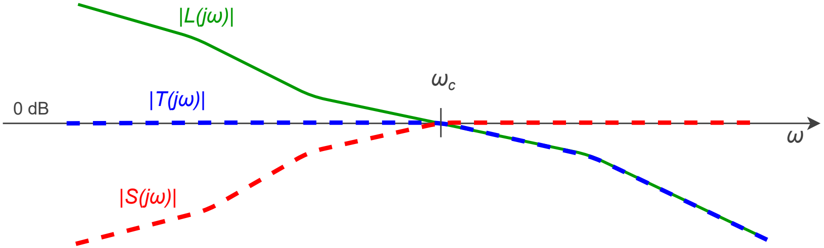
$$L(s) = P(s)C(s),$$

$$S(s) = \frac{1}{1+L(s)} = \frac{Y(s)}{D_{ff}(s)}, \quad T(s) = \frac{L(s)}{1+L(s)} = \frac{Y(s)}{W(s)} = -\frac{Y(s)}{D_{fb}(s)},$$

$$Q(s) = \frac{C(s)}{1+L(s)} = \frac{U(s)}{W(s)} = -\frac{U(s)}{D_{ff}(s)} = -\frac{U(s)}{D_{fb}(s)}$$

Foreword

The addressed *scenario* (in the Bode applicability hypotheses)

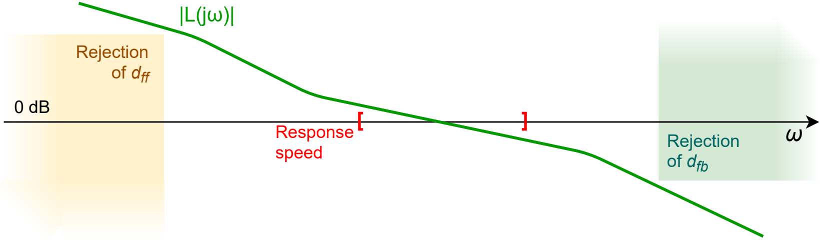


- Approximations for the frequency responses of $S(s)$ and $T(s)$:

$$S(j\omega) \approx \begin{cases} 1/L(j\omega) & \omega < \omega_c \\ 1 & \omega > \omega_c \end{cases} \qquad T(j\omega) \approx \begin{cases} 1 & \omega < \omega_c \\ L(j\omega) & \omega > \omega_c \end{cases}$$

Foreword

The addressed *scenario* (in the Bode applicability hypotheses)



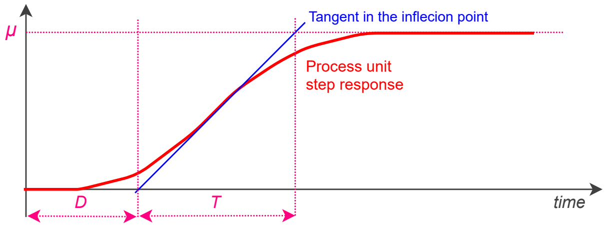
- Synthesis by loopshaping: determine type and/or gain of $C(s)$ based on steady-state requirements, set constraints as shown above, choose a suitable $L(s)$ avoiding critical cancellations and checking the phase margin, compute $C(s)$ dividing that $L(s)$ by $P(s)$. This, you know.
- We are looking for less powerful but simpler techniques.

Rapid PI/PID tuning for asymptotically stable overdamped processes

- Idea: approximate the system to control with a 1st or 2nd-order one, possibly with delay (FOPDT/SOPDT, for First/Second Order Plus Dead Time).
- Check out the literature for the said models and PI/PID tuning; here we stick to the FOPDT structure

$$M(s) = \mu \frac{e^{-sD}}{1+sT}, \qquad T > 0, D \geq 0.$$

- For an overdamped system one can readily parametrise it from a step response. There are many methods for that, we see only the inflection point one:



The simplest technique

Tune by cancellation

- Dominantly first-order system, negligible delay $\Rightarrow$ PI:

$$M(s)=\frac{\mu}{1+sT}, \text{ desired } \omega_c \quad \Rightarrow \quad C(s)=\frac{\omega_c}{s}\frac{1+sT}{\mu}$$

  giving a phase margin $\varphi_m \approx 90^\circ$ (90° in nominal conditions, i.e., $M=P$).

- Dominantly second-order system, negligible delay $\Rightarrow$ real PID:

$$M(s)=\frac{\mu}{(1+sT_1)(1+sT_2)}, \text{ desired } \omega_c \quad \Rightarrow \quad C(s)=\frac{\omega_c}{s}\frac{(1+sT_1)(1+sT_2)}{\mu\left(1+\frac{s}{\beta\omega_c}\right)}$$

  giving in nominal conditions $\varphi_m = 90^\circ - \arctan^\circ(1/\beta)$.

- Can easily extend to models with delay by constraining φ_m.
- However...

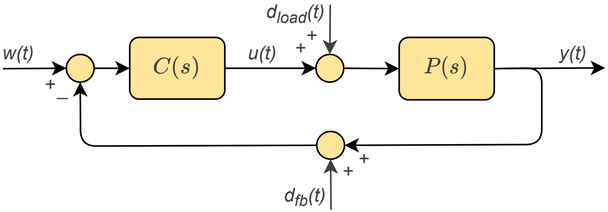
Accounting for load disturbances

- Abstract from F.G. Shinskey, "Process control: as taught vs. as practiced", Industrial & Engineering Chemistry Research **41**(16), 2002, 3745–3750, DOI `https://doi.org/10.1021/ie010645n`

  After 47 years of industrial experience and regular reviews of technical papers submitted to this and other journals, the author has discovered a number of common threads in papers written by academics that are at odds with industrial practice in process control. These indicators of the university-industry gap are so consistent over the years as to indicate that the gap is not closing. They include the study of loops with minimum-phase dynamics, overemphasis on set point response to the near exclusion of load regulation, omission of dynamics in the load path when covered at all, unrealistic economic considerations in objective functions, and a preference toward model-based control over PID control. Process models tend to be oversimplified, with little attention given to mass and energy balances and the bilinear models that they produce. Similarly, distributed dynamic models are rarely discussed, despite their predominance in mass and heat transfer. All this has led to research results left unused by industry and graduates left unprepared for industrial assignments.

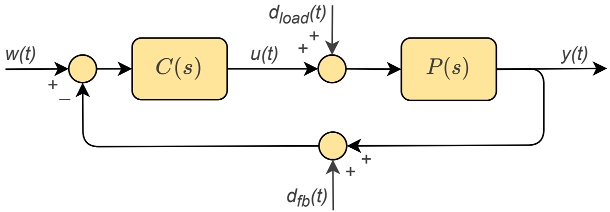
- Let us attempt to provide a small contribution toward filling the evidenced gap ☺

Accounting for load disturbances



- Load disturbances enter the loop at the process input.
- Also called “matched” disturbances as homogeneous to u.
- Frequent in process control and in particular in energy systems (e.g., u is cooling power for a room and d_{load} a heat rate from inhabitants/equipment in that room, or u and d_{load} are produced & consumed power in frequency control, as we shall see).

Accounting for load disturbances



- Transfer function from d_{load} to y:

$$G_{yd}(s) = \frac{P(s)}{1+L(s)}.$$

- Remark 1: does not depend only on $L(s)$.

  WARNING: $L(s)$ is adimensional (magnitude $\gtreqless 1$ makes sense) while $C(s)$ and $P(s)$ in general are not, hence units do count.

Accounting for load disturbances

- We know that $S(s)+T(s)=1$

  $\Rightarrow$ feedback cannot reject d_{ff} and d_{fb} in the same band.
- Here we have $Q(s)G_{yd}(s)=L(s)/(1+L(s))^2$,

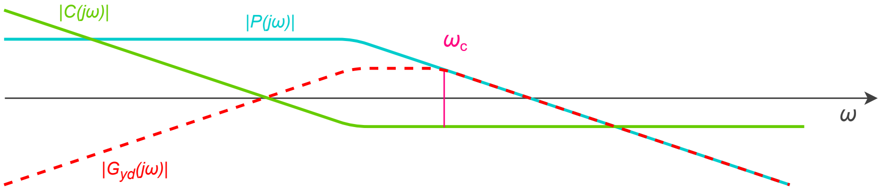
  $\Rightarrow$ to reject d_{load} by feedback in the vicinity of ω_c, where $|L|\approx 1$, $|Q|$ must be "large" in that band — i.e., we say, "feedback must be strong".
- Moreover, we can approximately write

$$Q(j\omega)\approx\begin{cases}1/P(j\omega) & \omega<\omega_c\\ C(j\omega) & \omega>\omega_c\end{cases}\qquad G_{yd}(j\omega)\approx\begin{cases}1/C(j\omega) & \omega<\omega_c\\ P(j\omega) & \omega>\omega_c\end{cases}$$

which further helps relating the rejection of d_{load} to the concept of "feedback strength".

Accounting for load disturbances

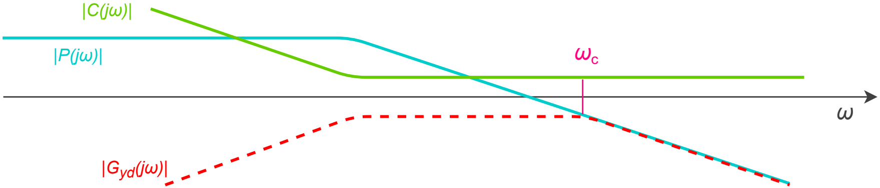
Weak feedback (illustrative example with 1st order process, PI controller)



- PI tuned by cancelling the process pole.
- The loop transfer function (not drawn for clarity) is ω_c/s, hence good set point tracking.
- However, poor rejection of d_{load}.

Accounting for load disturbances

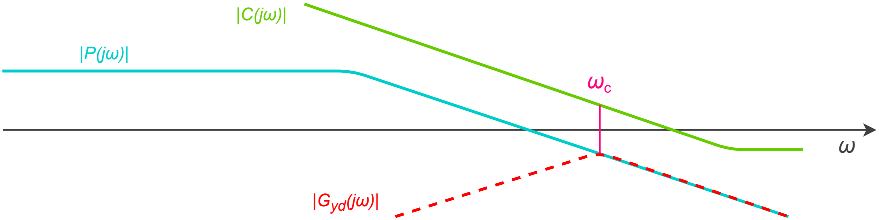
Strong feedback preserving cancellation tuning (same example)



- The loop transfer function is still ω_c/s but had to increase ω_c.
- Set point response can be too nervous (curable with 2-dof)...
- ...but worse, the high-frequency control sensitivity is large,
- hence measurement noise in d_{fb} is amplified on u (actuators are not happy with that, in general).

Accounting for load disturbances

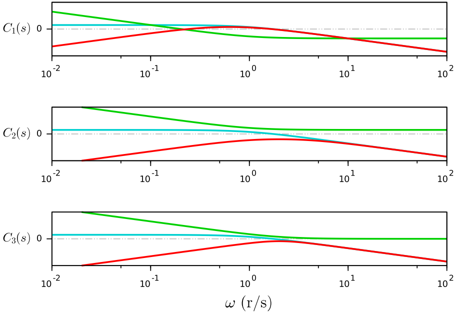
Strong feedback abandoning cancellation tuning (same example)

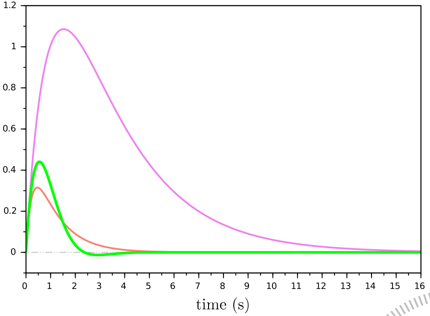


- Here $L(s)$ is not ω_c/s anymore (phase valley, watch stability).
- Set point response can be still recovered exploiting 2-dof...
- ...but this time the high-frequency control sensitivity is lower,
- and there is less/no *plateau* in $|G_{yd}(j\omega)|$, speeding up the rejection of d_{load} (more low-frequency content is cut).

Accounting for load disturbances

Bode plots and y responses to a d_{load} step – a numeric example



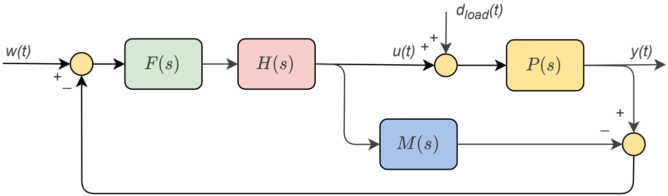


$$P(s) = \frac{2}{1+s};$$

$$C_1(s) = 0.2\left(1+\frac{1}{s}\right), \quad C_2(s) = 2\left(1+\frac{1}{s}\right), \quad C_3(s) = 1+\frac{1}{0.5s}.$$

A more formal and powerful technique

Internal Model Control scheme



- In nominal conditions ($M = P$, $d_{load} = 0$) the loop is open, hence

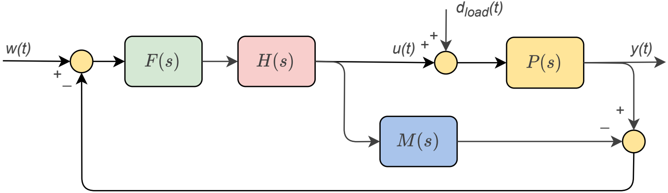
$$\frac{Y(s)}{W(s)} = P(s)H(s)F(s).$$

- And if in addition one could set $H(s) = P(s)^{-1}$,

$$\frac{Y(s)}{W(s)} = F(s).$$

A more formal and powerful technique

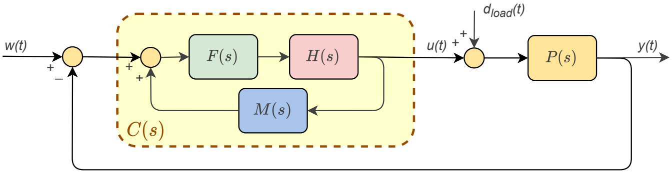
Internal Model Control scheme



- Elegant proof that
  - in the absence of model error and disturbances feedback is useless (the loop opens spontaneously),
  - one can assign the $w \rightarrow y$ dynamics, but with a relative degree at least equal to that of P (otherwise HF is not realisable)
  - and preserving in that dynamics possible non minimum-phase parts of P (otherwise, causality violation and/or critical cancellations).

A more formal and powerful technique

Internal Model Control (IMC) PI/PID tuning



- The IMC scheme is equivalent to a standard feedback one with

$$C(s) = \frac{H(s)F(s)}{1 - H(s)F(s)M(s)}.$$

- This is useful for specialising the scheme, e.g. to PI/PID controllers and FOPDT (asymptotically stable) models.

A more formal and powerful technique

Internal Model Control (IMC) PI/PID tuning

- Let us take

$$M(s) = \mu \frac{e^{-sD}}{1+sT}, \quad H(s) = \frac{1+sT}{\mu}, \quad F(s) = \frac{1}{1+s\lambda}.$$

- This corresponds to zero steady-state error for constant inputs as $F(0) = 1$.
- Parameter λ is readily interpreted as the desired closed-loop (dominant) time constant (easy to relate to a response speed request).
- Doing so we get

$$C(s) = \frac{\frac{1+sT}{\mu}\,\frac{1}{1+s\lambda}}{1-\frac{1+sT}{\mu}\,\frac{1}{1+s\lambda}\,\mu\,\frac{e^{-sD}}{1+sT}} = \frac{1}{\mu}\,\frac{1+sT}{1+s\lambda-e^{-sD}}.$$

A more formal and powerful technique

Internal Model Control (IMC) PI/PID tuning

- Now, replace the delay term with the (1,0) Padé approximation $1-sD$:

$$C(s)=\frac{1}{\mu}\frac{1+sT}{\cancel{1}+s\lambda-\cancel{1}+sD}=\frac{1+sT}{s\mu(\lambda+D)} \qquad \Rightarrow \quad \text{PI.}$$

- Instead, use the (1,1) Padé $\frac{1-sD/2}{1+sD/2}$:

$$C(s)=\frac{1}{\mu}\frac{1+sT}{1+s\lambda-\frac{1-sD/2}{1+sD/2}}=\ldots=\frac{(1+sT)(1+sD/2)}{s\mu(\lambda+D)\left(1+s\frac{\lambda D}{2(\lambda+D)}\right)} \qquad \Rightarrow \quad \text{real PID.}$$

- For the 1-dof ISA PI

$$C(s)=K\left(1+\frac{1}{sT_i}\right)$$

we readily get the IMC tuning formulæ $K=\frac{T}{\mu(\lambda+D)}$, $T_i=T$.

A more formal and powerful technique

Internal Model Control (IMC) PI/PID tuning

- For the 1-dof ISA PID

$$C(s)=K\left(1+\frac{1}{sT_i}+\frac{sT_d}{1+s\frac{T_d}{N}}\right)$$

the resulting IMC tuning formulæ – in sequence – are

$$T_i=T+\frac{D^2}{2(\lambda+D)},\quad K=\frac{T_i}{\mu(\lambda+D)},$$

$$N=\frac{T(\lambda+D)}{\lambda T_i}-1,\quad T_d=\frac{\lambda DN}{2(\lambda+D)}.$$

A more formal and powerful technique

Internal Model Control (IMC) PI/PID tuning

- Easy to employ (check it out *by hand*).
- Useful for rapid adaptation: keep F, update M and C updates as a consequence.
- Capable of accommodating for a delay in the model.
- Capable of tuning also the "derivative pole" of the real PID. Tuning an ideal PID and then setting N to a "default high" value can displace its zeroes to undesired positions; recall also that the high-frequency Q magnitude is $K(1+N)$.

- However still operating by cancellation, with the known shortcomings as for the rejection of d_{load}.

A more formal and powerful technique

S-IMC PI tuning

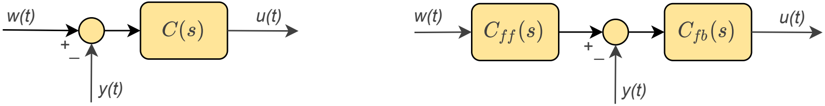
- Remember the example? We pushed the PI zero to a higher frequency w.r.t. the model pole.
- That is, we attempted to make T_i as small as possible compatibly with the specifications.
- S-IMC rule (we can limit the scope to the PI case):

$$K = \frac{T}{\mu(\lambda + D)}, \quad T_i = \min\left(T, 4(\lambda + D)\right).$$

- Same interpretation for λ.
- Check it out for d_{load} rejection in comparison to standard IMC.

One- and two-degree-of-freedom (1-dof and 2-dof) controllers

more formally



- 1-dof (left): $\frac{U(s)}{W(s)} = -\frac{U(s)}{Y(s)} = C(s)$; cannot assign the two independently.
- 2-dof (right): $\frac{U(s)}{W(s)} = C_{fb}(s)C_{ff}(s)$, $\frac{U(s)}{Y(s)} = C_{fb}(s)$; can assign the two independently.
- In the 2-dof case $C_{ff}(s)$ acts as a set point filter, hence must be asymptotically stable and with unity gain.
- Apparently, only $C_{fb}(s)$ influences the loop stability.

Regulatory *versus* servo tuning

in general

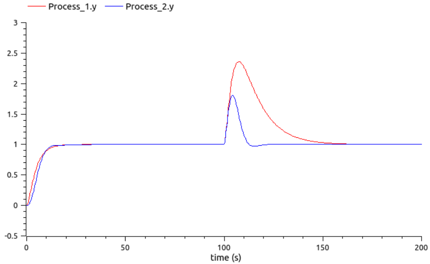
- Regulatory tuning
  primary purpose is disturbance rejection, strong feedback is almost invariantly required, only C_{fb} is involved (stability/rejection tradeoff).
- Servo tuning
  primary purpose is set point tracking, both C_{fb} and C_{ff} are involved.

- There are tuning rules for C_{fb} to privilege either objective.
- It is common practice to first tune C_{fb} for stability and disturbance rejection, then use C_{ff} to recover set point tracking if required.
- Remember that feedback kills uncertainty only up to ω_c, hence beware if C_{ff} needs to accelerate the system beyond (unless your process model is guaranteed good enough in that band).

Exploiting 2-dof controllers

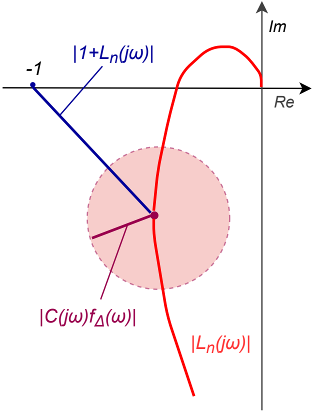
An example

- Process:
$$P(s) = \frac{1}{(1+10s)(1+s)}.$$
- Weak and strong feedback:
$$C_{fb1} = 2\left(1+\frac{1}{10s}\right),\ C_{fb2} = 4\left(1+\frac{1}{4s}\right).$$
- Feedforward (set point prefilter):
$$C_{ff1} = 1, \quad C_{fb2} = \frac{1}{1+5s}.$$
- Set point and load disturbance unit step: observe the controlled variable response.



Quantifying robustness *a priori*

A sufficient condition for robust stability (in the Bode hypotheses)



- Nominal loop $L_n(j\omega)$ does not encircle -1.
- Additive model error: $P(s) = M(s) + \Delta_P(s)$.
- Loop transfer function: $L(s) = (M(s) + \Delta_P(s))C(s)$.
- That is, w.r.t. nominal, $L(s) = L_n(s) + \Delta_P(s)C(s)$.
- Error magnitude bound: $|\Delta_P(j\omega)| < f_\Delta(\omega)\ \forall\omega$.
- Sufficient condition: $|C(j\omega)f_\Delta(\omega)| < |1 + L_n(j\omega)|\ \forall\omega$.
- Nominal control sensitivity: $Q_n(s) = \frac{C(s)}{1+L_n(s)}$.
- Summing up, the condition is

$$f_\Delta(\omega) < \frac{1}{Q_n(j\omega)} \quad \forall\omega.$$

The inverse of the nominal control sensitivity provides a magnitude bound for the acceptable additive model error.

Main control structures for energy systems

Feedforward compensation

Cascade control

Decoupling

Smith predictor

Major actuation schemes

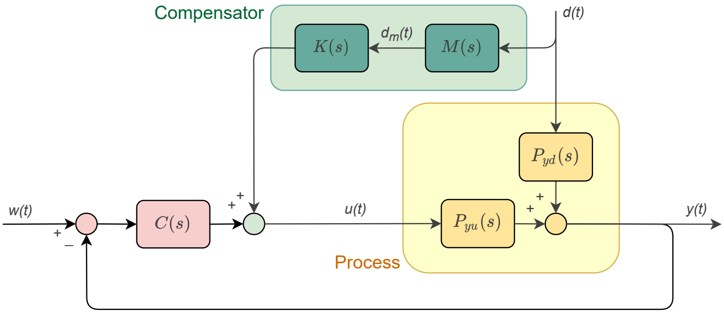
Foreword

- We shall now review the major control structures of interest for us, i.e.,
  - feedforward compensation,
  - cascade control,
  - multivariable control with decoupling,
  - Smith predictor
- We shall also review some relevant actuation schemes, i.e.,
  - split range,
  - daisy-chaining,
  - time division output.

Foreword

- In this part of the course we shall centre our treatise on the modulating side of the matter and stay in an LTI context.
- It is however worth noticing right from now that implementing control structures often requires to set up some logic as well, and in certain cases also a proper management of nonlinear controller features like antiwindup.
- We are dealing with these "advanced" aspects – to a depth compatible with the course – later on, when we shall apply the structures that we are now revising to specific (energy-related) cases.

Feedforward compensation



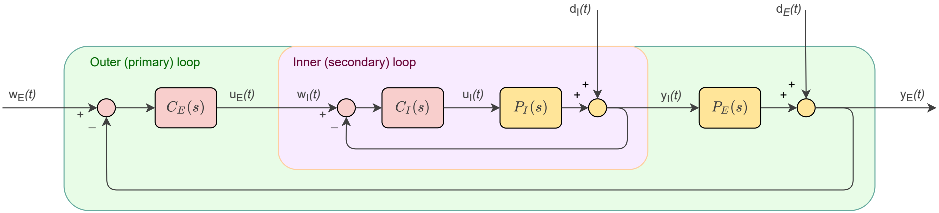
- Purpose: reduce the influence on the controlled variable $y(t)$ of a measurable disturbance $d(t)$ acting on the loop forward path.
- How: by computing $K(s)$ so that $Y(s)/D(s)$ be ideally zero, i.e.,

$$\frac{P_{yd}(s)+M(s)K(s)P_{yu}(s)}{1+C(s)P_{yu}(s)}=0, \quad \Rightarrow K_{ID}(s)=-\frac{P_{yd}(s)}{M(s)P_{yu}(s)}$$

Feedforward compensation

- The so found compensator is termed "ideal" (ID subscript) as
  - it may have more zeroes than poles (thus not being realisable),
  - and/or have RHP poles (thus producing critical cancellations).
- In such cases one has to obtain from $K_{ID}(s)$ the real compensator $K(s)$
  - omitting zeroes and/or adding poles,
  - and in any case not introducing RHP poles.
- This will yield a compensation "valid up to a certain frequency", namely that for which $K(j\omega)$ starts to differ "significantly" from $K_{ID}(j\omega)$, accounting for *both* magnitude and phase.

Cascade control

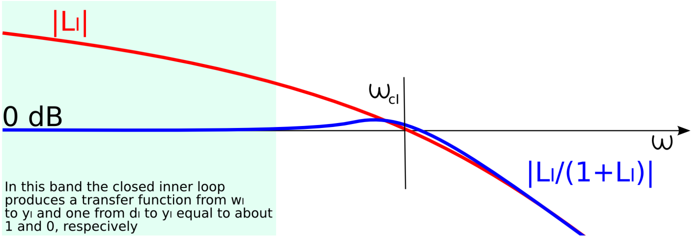


- Purpose: mitigate the effects of a disturbance $d_I(t)$ the effects of which appear on some measurable process variable $y_I(t)$ "before" – in a dynamic sense – they eventually show up on the primary controlled variable $y_E(t)$.
- How: by closing a "fast" inner (secondary) loop so as to hide both the dynamics of P_I and the effects of d_I to the outer (primary) one.
- Two remarks:
  - the scheme has the inherent cost of measuring $y_I(t)$,
  - and is hardly of any use (i.e., a single-loop control could do the job as well) in the absence of a significant $d_I(t)$.

Cascade control

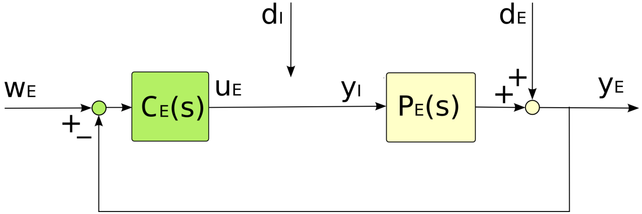
Explanation in the frequency domain

- Denote by $L_I(s) = C_I(s)P_I(s)$ the inner loop transfer function:



- In practice, indicating by ω_{cI} and ω_{cE} the critical frequencies of the inner and the outer loop, respectively, a minimum bandwidth separation of 0.5–1 decade is advised.
- If this is accomplished, it is possible to (approximately but reliably) compute ω_{cE} as if the outer (open) loop transfer function were just given by $C_E(s)P_E(s)$.

Cascade control



- For the synthesis, therefore, one can view the outer loop as in the block diagram above.
- Overall, this leads to determining
  - $C_I(s)$ based on $P_I(s)$ only
  - and $C_E(s)$ based on $P_E(s)$ only (preserving however the required band separation).

Multivarlable control with decoupling

- Purpose: address Multiple Input, Multiple Output (MIMO) processes that are
  - square, i.e., with as many control inputs as controlled variables,
  - and interacting, i.e., where a control input influences more than one of the controlled variables.
- How: by taking a two-step approach, namely
  - by first prepending to the process a decoupler MIMO block so that the cascade of the two be diagonal,
  - and then closing one SISO loop per variable, synthesised with the known techniques.
- Remarks:
  - we treat the 2×2 scheme, generalising to $n \times n$ is straightforward;
  - there exist formal techniques for pairing, i.e. deciding which control input to use for governing which controlled variable; here we do not treat such techniques, and assume that pairing comes from physical considerations.

Multivarlable control with decoupling

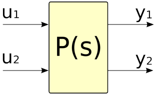
- A 2×2 MIMO process is described in the LTI framework as

$$\begin{bmatrix} Y_1(s) \\ Y_2(s) \end{bmatrix} = \begin{bmatrix} P_{11}(s) & P_{12}(s) \\ P_{21}(s) & P_{22}(s) \end{bmatrix} \begin{bmatrix} U_1(s) \\ U_2(s) \end{bmatrix}$$

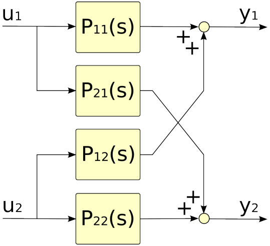
  i.e., by the transfer matrix

$$P(s) = \begin{bmatrix} P_{11}(s) & P_{12}(s) \\ P_{21}(s) & P_{22}(s) \end{bmatrix}$$

- In terms of block diagrams this means



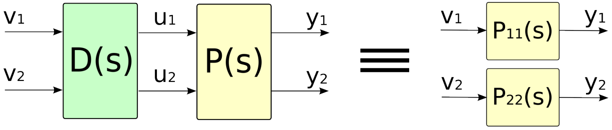
$$\begin{cases} Y_1 = P_{11}U_1 + P_{12}U_2 \\ Y_2 = P_{21}U_1 + P_{22}U_2 \end{cases} \Rightarrow$$



Multivarlable control with decoupling

The decoupling block

- The purpose of $D(s)$ is to realise the equivalence indicated below:



i.e., setting $Y' = [Y_1\, Y_2]$, $U' = [U_1\, U_2]$ e $V' = [V_1\, V_2]$,

$$Y = PU = PDV = \begin{bmatrix} P_{11}(s) & 0 \\ 0 & P_{22}(s) \end{bmatrix} V$$

- Therefore, $D(s)$ is determined as

$$D(s) = P^{-1}(s) \begin{bmatrix} P_{11}(s) & 0 \\ 0 & P_{22}(s) \end{bmatrix}$$

Multivarlable control with decoupling

The decoupling block

- Interpretation:

$$D = \begin{bmatrix} P_{11} & P_{12} \\ P_{21} & P_{22} \end{bmatrix}^{-1} \begin{bmatrix} P_{11} & 0 \\ 0 & P_{22} \end{bmatrix}$$

hence

$$\begin{aligned} D^{-1} &= \left(\begin{bmatrix} P_{11} & P_{12} \\ P_{21} & P_{22} \end{bmatrix}^{-1} \begin{bmatrix} P_{11} & 0 \\ 0 & P_{22} \end{bmatrix} \right)^{-1} \\ &= \begin{bmatrix} P_{11} & 0 \\ 0 & P_{22} \end{bmatrix}^{-1} \begin{bmatrix} P_{11} & P_{12} \\ P_{21} & P_{22} \end{bmatrix} \\ &= \frac{1}{P_{11}P_{22}} \begin{bmatrix} P_{22} & 0 \\ 0 & P_{11} \end{bmatrix} \begin{bmatrix} P_{11} & P_{12} \\ P_{21} & P_{22} \end{bmatrix} \\ &= \frac{1}{P_{11}P_{22}} \begin{bmatrix} P_{11}P_{22} & P_{12}P_{22} \\ P_{11}P_{21} & P_{11}P_{22} \end{bmatrix} = \begin{bmatrix} 1 & P_{12}/P_{11} \\ P_{21}/P_{22} & 1 \end{bmatrix} \end{aligned}$$

Multivarlable control with decoupling

The decoupling block

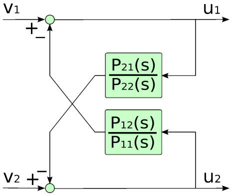
- Carrying on, $U = DV \Rightarrow V = D^{-1}U$, thus

$$\begin{bmatrix} V_1(s) \\ V_2(s) \end{bmatrix} = \begin{bmatrix} 1 & \frac{P_{12}}{P_{11}} \\ \frac{P_{21}}{P_{22}} & 1 \end{bmatrix} \begin{bmatrix} U_1(s) \\ U_2(s) \end{bmatrix} \quad \Rightarrow \quad \begin{cases} V_1 & = & U_1 + \frac{P_{12}}{P_{11}} U_2 \\ V_2 & = & \frac{P_{21}}{P_{22}} U_1 + U_2 \end{cases}$$

- In synthesis, then

$$\begin{cases} U_1 & = & V_1 - \frac{P_{12}}{P_{11}} U_2 \\ U_2 & = & V_2 - \frac{P_{21}}{P_{22}} U_1 \end{cases}$$

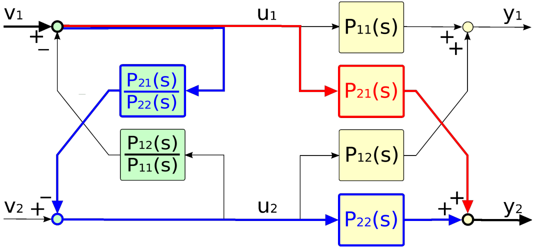
and the decoupler is described by the block diagram



whence the frequently encountered name "backward decoupling".

Multivarlable control with decoupling

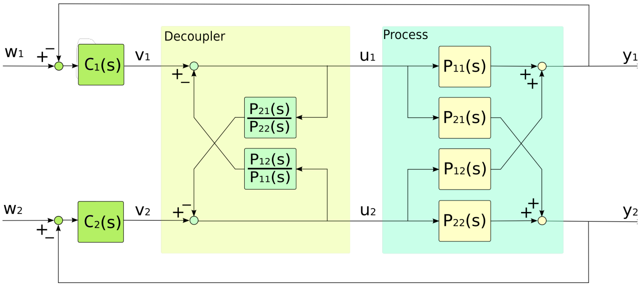
The decoupling block



- The scheme shows the backward decoupler operation by evidencing how it zeroes the net signal path from v_1 to y_2 (sum of read and blue); of course the same holds for the symmetric path.
- The same feasibility/stability issues of feedforward compensation may arise, requiring to use approximated decoupling blocks and thus limiting the band where decoupling is effective.

Multivarlable control with decoupling

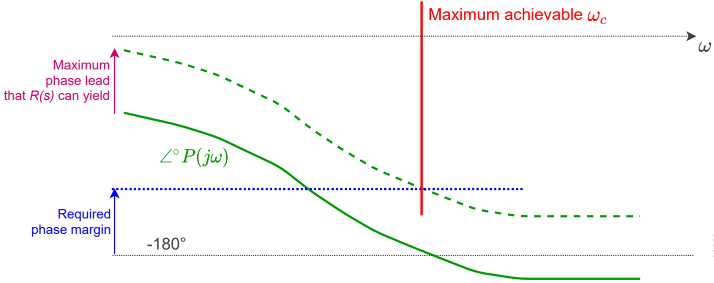
The overall scheme



- The two controllers $C_1(s)$ and $C_2(s)$ are designed with known SISO techniques, as if dealing with two independent processes having transfer functions $P_{11}(s)$ and $P_{22}(s)$, respectively.

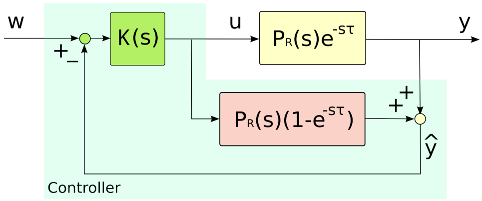
Smith predictor

- Purpose: address cases where the process has so large a delay that obtaining a certain stability degree (e.g., a desired phase margin) requires to reduce performance (e.g., response speed in terms of ω_c) unacceptably.
- Let us interpret this graphically:



- This is how one *meaningfully* says that a delay is "too large".

Smith predictor



- How: by observing that in the scheme above

$$\frac{\widehat{Y}(s)}{U(s)} = P_R(s)$$

where the transfer function $P_R(s)$ is assumed rational.

- Block $K(s)$ can thus be synthesised with known methods, accounting only for the rational dynamics of the process.

Smith predictor

- The scheme immediately yields

$$\frac{\widehat{Y}(s)}{W(s)} = \frac{K(s)P_R(s)}{1+C(s)P_R(s)}$$

- Additionally, the "controller" in the same scheme is equivalent to a feedback one with transfer function

$$C(s) = \frac{K(s)}{1+K(s)P_R(s)(1-e^{-s\tau})}$$

thus

$$\begin{aligned}\frac{Y(s)}{W(s)} &= \frac{C(s)P_R(s)e^{-s\tau}}{1+C(s)P_R(s)e^{-s\tau}} \\ &= \frac{\frac{K(s)P_R(s)e^{-s\tau}}{1+K(s)P_R(s)(1-e^{-s\tau})}}{1+\frac{K(s)P_R(s)e^{-s\tau}}{1+K(s)P_R(s)(1-e^{-s\tau})}} = \ldots = \frac{K(s)P_R(s)e^{-s\tau}}{1+K(s)P_R(s)}\end{aligned}$$

Smith predictor

- Summing up, therefore, we get

$$\frac{Y(s)}{W(s)} = \frac{\widehat{Y}(s)}{W(s)} e^{-s\tau}$$

which means that synthesising $K(s)$ based on $P_R(s)$ and using the Smith predictor scheme, the obtained behaviour of y is the same as that of $\widehat{y}$, just delayed by τ.

- Caveat:
  - the model has to be "more precise" than is needed for mere feedback control
  - and disturbances need not to be too significant (ore be *very well* compensated for in the prediction path)

otherwise $\widehat{y}$ ceases to be a good prediction of y, to the detriment of the scheme operation.

Actuation schemes

General idea

- In general we think of schemes in which a controller drives one actuator.
- In several cases, this is not really true.
- Sometimes an action as decided by a controller has to be exerted through more articulated physical machineries...
- ...that however *at the control synthesis level* need representing in such a way to accept one "command" from the controller.
- Dealing with this part of a control system means talking about actuation schemes.
- We are going to see a few ones of interest for us.

Split range

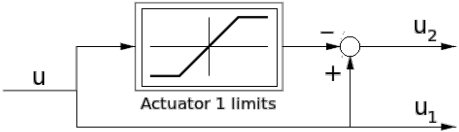
- Purpose: make two actuators behave like a single one by having each of them act in a different range of the control variable (whence the name).
- Typical example: a temperature controller with one actuator for heating for heating and one for cooling.
- How: denoting by $u_1 \in [0,1]$ and $u_2 \in [0,1]$ the two actuators and supposing – without loss of generality – that the transition happens for $u=0$, where $u \in [-1,1]$ is the controller output, by simply setting

$$u_1 = \begin{cases} u & u \geq 0 \\ 0 & u < 0 \end{cases} \qquad u_2 = \begin{cases} 0 & 0 \geq 0 \\ -u & u < 0 \end{cases}$$

- Note: sometimes a *dead zone* is introduced around $u=0$ to avoid switching in and out the two actuators too frequently.

Daisy chaining

- Purpose: have several actuators activated in sequence (the $i+1$-th starting to operate when the i-th has reached its maximum).
- Most typical motivation: start with the most energy-efficient actuator (e.g., a heat pump) and have a less efficient one (e.g., an electric heater) intervene only if the first one is not sufficient.
- How: by using the diagram



that easily generalises to an arbitrary number of actuators.

- Note:here too dead zones are sometimes introduced, for the same reason mentioned above.

Time Division Output

- Purpose: make an on/off actuator behave like a modulating one.
- Most typical motivations:
  - modulating high-power actuators may be impractical or even impossible;
  - even in the absence of the above problem, operating the actuator not at 100% may reduce its efficiency.
- How: by deciding an *actuation period* T_a, small w.r.t. the process dynamics' time scale, taking this as the sampling time for the (digital) controller, and having the control signal $u \in [0,1]$ provide the actuator activation's duty cycle.

- Example: with $T_a = 10\,s$, $u = 0.6$ means that in the sampling period the actuator will be on for $6\,s$ and then off for $4\,s$.

Part B

Electric systems

Electric systems - component models

Overview – models and control problems

Generators – thermoelectric case

Controlled generators – still on the thermo case

Generalising (a bit) to other generator types

Lines and loads

Models and control problems

in the "traditional" *scenario*

- In traditional control of AC electric networks, two types of problems are encountered:
  - **power and cost control**, i.e.,
    deliver the required power to all utilisers while
    - minimising costs (and possibly emissions) *globally* (general interest),
    - and/or maximising the economic revenue *for one or a set of generators* (particular interest),

    which are in general conflicting objectives;
  - **energy quality control**, i.e.,
    deliver electricity at the required voltage and frequency, which necessarily involves cooperation as all the generators (and loads) are coupled by the network, hence no particular interest makes sense.
- The above problems are clearly intertwined, in a way that however depends on the generator operation.
- Also, in the considered *scenario* only generators act to provide control: the power demand by loads is considered exogenous (i.e., a disturbance).

Models and control problems

in the "traditional " *scenario*

- In this respect, two generator types are in fact distinguished:
  - with rotating masses, i.e., with an alternator (e.g. thermo, nuclear, hydro, wind, tidal, thermal – or "thermodynamic" – solar),
  - and without rotating masses, i.e., with an inverter (e.g. photovoltaic solar).
- Rotating masses inherently couple power and frequency, as a power excess or deficiency accelerates or decelerates the masses, thereby altering frequency.
- This does not hold true when no rotating mass is present.
- Other problems exist that do not fit in the space of this course, like e.g. reactive power control.
- Since we aim at system-level principles and models, we shall examine generators having the thermoelectric case as reference, and then generalise to a feasible extent.
- Our KPIs will be power and frequency error, generation cost, revenue, and transmission losses.

Models and control problems

in the "traditional" *scenario*

- When moving from generation to network control, another relevant problem is encountered:
  - **load flow**, i.e., delivering power without overloading transmission lines, and possibly minimising line losses.
- Load flow can just provide constraints for power and quality control to avoid overloads (and we shall just say some words on this) or be part of the overall optimisation, leading to the *optimal flow* problem that we cannot treat here.
- Finally, as we shall see soon, power and quality problems require to abstract different generator and network element interfaces.
- Let us now proceed to model generators: as anticipated, we start from the thermoelectric case and then extend the ideas.

Thermoelectric generators

Basics (in a view to power and frequency control)

- These generators include a rotating mass, whence the power/frequency coupling.
- Synthesis of operation:
  - fuel burns in a *furnace* and produces heat;
  - heat turns water into superheated steam;
  - steam moves a turbine;
  - the turbine moves the alternator,
- The generator has its internal controls, which we do not represent,
- and we also disregard the details about the water/steam path (free or forced circulation, once-through)...
- ...since the matter is treated in dedicated courses, and is not to be represented in our system-level models.

Thermoelectric generators

A system-level model – prime mover

- We start from the *prime mover*, i.e., the system having fuel as inlet and mechanical power for the alternator as outlet.
- For simplicity we take as exogenous input the combustion power P_c released to the *main energy storage* (steam); fuel consumption will come into play later on.
- In so doing we neglect heat storage in the combustion chamber and the flue gas path as they are very small w.r.t. those in the metal and water/steam path, which we assume thermally coherent.
- The stored energy balance is thus

$$\dot{E} = P_c - P_{loss} - P_t$$

where P_{loss} is the power lost to the external environment and P_t the power drawn by the turbine.

Thermoelectric generators

A system-level model – prime mover

- We simplistically assume that the main energy storage is composed of saturated steam and its mass is constant; we thus relate P_{loss} to the difference between the saturation temperature at the steam pressure p (which thereby comes to represent the stored energy) and the external temperature, as

$$P_{loss} = G_{loss}(T_{sat}(p) - T_{ext})$$

  where G_{loss} is an equivalent thermal conductance.
- Even more simplistically, since in general $T_{sat}(p) \gg T_{ext}$, we write

$$P_{loss} = K_{loss}E/M$$

  where K_{loss} is a parameter and the division by M represents the fact that P_{loss} depends on the steam *specific state*; K_{loss} is of course related to the dispersing surface.

Thermoelectric generators

A system-level model – prime mover

- We now make another simplistic assumption by disregarding the superheating that steam undergoes prior to traversing the turbine valve, and assume P_t to depend on the steam pressure (that is, on E/M) and the turbine valve opening $\theta \in [0,1]$ as

$$P_t = \theta K_{draw} E/M$$

  where K_{draw} is another parameter.
- The mechanical power to the alternator is then obtained by accounting for a mechanical efficiency, assumed constant and denoted by η_m, as

$$P_m = \eta_m P_t.$$

Thermoelectric generators

A system-level model – prime mover

- Putting it all together, we have

$$\begin{cases} \dot{E} &=& P_c - K_{loss}E/M - \theta K_{draw}E/M \\ P_m &=& \eta_m \theta K_{draw}E/M \end{cases}$$

- Note that $[K_{loss}E/M] = [K_{draw}E/M] = [W]$ since θ and η_m are adimensional, thus $[K_{loss}/M] = [K_{draw}/M] = [1/s]$ and we can write

$$\begin{cases} \dot{E} &=& P_c - \left(\dfrac{1}{T_{loss}} + \dfrac{\theta}{T_{draw}}\right) E \\ P_m &=& \dfrac{\eta_m}{T_{draw}} E\theta \end{cases}$$

- In the state equation above, T_{loss} and T_{draw} are interpreted as the time constants with which energy is lost into the environment and drawn to the alternator at full throttling (turbine) valve opening.

Thermoelectric generators

A system-level model – prime mover

- Notice that the plant size, intuitively indicated e.g. by (some nominal value for) the contained water/steam mass M, in this formulation comes to be represented by the introduced time constants (larger plant, larger M, larger T_{loss} and T_{draw}).

- Of course models like this one are in the large *extremely* coarse, and for real-life uses they could only have local validity around an operating point.
- Treating such aspects strays from the course, however, and therefore we shall just use the model *as is* since this is more than enough for our purposes.

Thermoelectric generators

A system-level model – prime mover

- We can furthermore introduce a generator nominal power P_n and denote by T_{rest} the time required to restore the generator from zero to its "nominal energy storage" E_n operating at nominal power, thereby defining E_n as $P_n T_{rest}$. Dividing the above equations by E_n we get

$$\begin{cases} \dot{e}_n = \dfrac{1}{T_{rest}} p_c - \left(\dfrac{1}{T_{loss}} + \dfrac{\theta}{T_{draw}} \right) e_n \\ p_m = \eta_m \dfrac{T_{rest}}{T_{draw}} e_n \, \theta \end{cases}$$

where $p_c = P_c/P_n$ and $p_m = P_m/P_n$ are respectively the normalised combustion and mechanical powers, while $e_n = E/E_n$ is the normalised energy storage (*not* the steam specific energy, beware).

Thermoelectric generators

A system-level model – prime mover

- As such, indicating – for this discussion only – the steam specific energy $[J/kg]$ with e_{spec}, we could better rewrite the second model equation (mind the different meanings of e_{spec} and e_n) as

$$P_m = \eta_m K_{draw} e_{spec}\,\theta = \eta_m K_{draw}\frac{E}{M}\,\theta = \eta_m K_{draw}\frac{P_n T_{rest}}{M} e_n\,\theta$$

- Thus, normalising by P_n,

$$p_m = \eta_m K_{draw}\frac{T_{rest}}{M} e_n\,\theta$$

- Overall, the quantity $k_\theta := K_{draw}T_{rest}/M$ acts as sort of a "valve gain" cascaded to the mechanical efficiency. Investigating its role would however require relating also the contained *mass* to the energy state, which we do not want to do for our purposes.

Thermoelectric generators

A system-level model – prime mover

- Thus, we shall simply write

$$\begin{cases} \dot{e}_n = \dfrac{1}{T_{rest}} p_c - \left(\dfrac{1}{T_{loss}} + \dfrac{\theta}{T_{draw}} \right) e_n \\ p_m = \eta_m k_\theta \, e_n \, \theta \end{cases}$$

  and for simplicity (without didactic loss for this course) assume $k_\theta = 1$, hence omitting it hereinafter.

- Summing up, we shall take as model for the generator prime mover the dynamic system

$$\begin{cases} \dot{e}_n = \dfrac{1}{T_{rest}} p_c - \left(\dfrac{1}{T_{loss}} + \dfrac{\theta}{T_{draw}} \right) e_n \\ p_m = \eta_m \, e_n \, \theta \end{cases}$$

Thermoelectric generators

A system-level model – prime mover

- We can now determine the equilibrium of the previous model for constant inputs $\overline{p}_c, \overline{\theta}$ (the overline denotes equilibrium values) using e.g. the Maxima script

```
/* Model
*/
endot : pc/Trest-(1/Tloss+theta/Tdraw)*en;
pm    : etam*en*theta;

/* Equilibrium
*/
enbar : rhs(solve(subst([pc=pcbar,theta=thetabar],endot),en)[1]);
pmbar : ratsimp(etam*enbar*thetabar);
```

- This produces

$$\overline{e}_n = \frac{T_{draw}T_{loss}\overline{p}_c}{T_{rest}\left(T_{draw}+T_{loss}\overline{\theta}\right)}, \quad \overline{p}_m = \eta_m \overline{e}\,\overline{\theta}.$$

Thermoelectric generators

A system-level model – prime mover

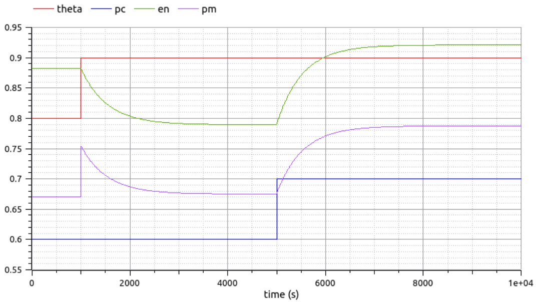
- We can also simulate the prime mover model in Modelica, starting at the equilibrium and applying steps to θ and p_c, with

```
model SimpleThermoElecGenPM
  parameter Real Pn       = 100;
  parameter Real Trest    = 400;
  parameter Real Tdraw    = 500;
  parameter Real Tloss    = 1e4;
  parameter Real etam     = 0.95;
  parameter Real thetabar = 0.8;
  parameter Real pcbar    = 0.6;
  Real en(start=Tdraw*Tloss*pcbar/Trest/(Tdraw+Tloss*thetabar));
  Real pc,pm,theta;
equation
  der(en) = pc/Trest-(1/Tloss+theta/Tdraw)*en;
  pm      = etam*en*theta;
  theta   = if time<1000 then thetabar else thetabar+0.1;
  pc      = if time<5000 then pcbar    else pcbar+0.1;
end SimpleThermoElecGenPM;
```

Thermoelectric generators

A system-level model – prime mover

- Simulating for 10000 s produces



- The θ step ($t = 1000\,s$) yields a sudden p_m response but then, as p_c is constant, p_m settles back while e_n decreases and settles as well, both transients being dominated by the storage time constant.
- The p_c step ($t = 5000\,s$) makes both p_m and e_n increase and settle, however with the storage time scale (no sudden p_m response).

Thermoelectric generators

A system-level model – prime mover

- We can furthermore linearise the model in the vicinity of the equilibrium, setting $\Delta p_c = p_c - \overline{p}_c$, $\Delta\theta = \theta - \overline{\theta}$ and $\Delta e = e - \overline{e}$, $\Delta p_m = p_m - \overline{p}_m$, and taking as outputs both Δp_m and Δe:

```
/* Linearised model state space matrices
*/
A     : subst([pc=pcbar,theta=thetabar,en=enbar],jacobian([endot],[en]));
B     : subst([pc=pcbar,theta=thetabar,en=enbar],jacobian([endot],[theta,pc]));
C     : subst([pc=pcbar,theta=thetabar,en=enbar],jacobian([pm,en],[en]));
D     : subst([pc=pcbar,theta=thetabar,en=enbar],jacobian([pm,en],[theta,pc]));
```

- The result is

$$
\begin{cases}
\Delta\dot{e}_n = -\left(\dfrac{1}{T_{loss}} + \dfrac{\overline{\theta}}{T_{draw}}\right)\Delta e_n + \begin{bmatrix} -\dfrac{\overline{p}_c T_{loss}}{T_{rest}(T_{draw}+T_{loss}\overline{\theta})} & \dfrac{1}{T_{rest}} \end{bmatrix} \begin{bmatrix} \Delta\theta \\ \Delta p_c \end{bmatrix} \\[2ex]
\begin{bmatrix} \Delta p_m \\ \Delta e_n \end{bmatrix} = \begin{bmatrix} \eta_m \overline{\theta} \\ 1 \end{bmatrix} \Delta e_n + \begin{bmatrix} \dfrac{\eta_m \overline{p}_c T_{draw} T_{loss}}{T_{rest}(T_{draw}+T_{loss}\overline{\theta})} & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \Delta\theta \\ \Delta p_c \end{bmatrix}
\end{cases}
$$

- Note that it is asymptotically stable, as T_{draw}, T_{loss} and $\overline{\theta}$ are all positive.

Thermoelectric generators

A system-level model – prime mover

- We can finally compute the transfer matrix of the linearised model:

```
/* Linearised model transfer matrix. Note that here A is actually scalar;
   the syntax for the matrix-A case would be C.invert(s*ident(size_of_a)-A).B+D
*/
Gamma : factor(C.B*invert(s-A)+D);
```

- We get

$$
\begin{aligned}
\Gamma(s) &= \begin{bmatrix} \Gamma_{\theta m}(s) & \Gamma_{cm}(s) \\ \Gamma_{\theta e}(s) & \Gamma_{ce}(s) \end{bmatrix} = \begin{bmatrix} \dfrac{\Delta p_m(s)}{\Delta\theta(s)} & \dfrac{\Delta p_m(s)}{\Delta p_c(s)} \\ \dfrac{\Delta e_n(s)}{\Delta\theta(s)} & \dfrac{\Delta e_n(s)}{\Delta p_c(s)} \end{bmatrix} = \ldots \\
&= \frac{1}{1+s\frac{T_{draw}T_{loss}}{T_{draw}+T_{loss}\overline{\theta}}} \begin{bmatrix} \frac{\eta_m T_{draw}^2 T_{loss}\overline{p}_c}{T_{rest}(T_{draw}+T_{loss}\overline{\theta})^2}(1+sT_{loss}) & \frac{\eta_m T_{draw}T_{loss}\overline{\theta}}{T_{rest}(T_{draw}+T_{loss}\overline{\theta})} \\ -\frac{T_{draw}T_{loss}^2\overline{p}_c}{T_{rest}(T_{draw}+T_{loss}\overline{\theta})^2} & \frac{T_{draw}T_{loss}}{T_{rest}(T_{draw}+T_{loss}\overline{\theta})} \end{bmatrix}
\end{aligned}
$$

- Note that all the elements have relative degree 1 except for $\Delta p_m/\Delta\theta$, which has 0; this is consistent with the simulated responses.

Thermoelectric generators

A system-level model – balance at the alternator

- In our *scenario* the demanded active power P_e is exogenous for the generator, hence the energy equation for the rotating mass (turbine and alternator) reads

$$J\omega\dot{\omega} = P_m - P_e$$

  where J is the total inertia seen at the shaft, and ω the angular velocity (which we identify with the electric frequency, disregarding for simplicity the number of alternator polar expansions).
- The equation above yields possible equilibria at any ω, provided that the (constant) values $\overline{P}_m$ and $\overline{P}_e$ coincide. Linearising, therefore,

$$\Delta\dot{\omega} = -\frac{P_m - P_e}{J\overline{\omega}^2}\Delta\omega + \frac{1}{J\overline{\omega}}(\Delta P_m - \Delta P_e)$$

- Assuming ω regulated at its desired value ω_o (we shall guarantee this shortly) and recalling that at the equilibrium $P_m = P_e$, we have

$$\Delta\dot{\omega} = \frac{1}{J\omega_o}(\Delta P_m - \Delta P_e)$$

Thermoelectric generators

A system-level model – balance at the alternator

- Normalising $P_{m,e}$ with P_n and ω with ω_o and using δ for the variations of normalised quantities, finally,

$$\frac{\Delta\dot{\omega}}{\omega_o P_n} = \frac{1}{J\omega_o}\left(\frac{\Delta P_m}{\omega_o P_n} - \frac{\Delta P_e}{\omega_o P_n}\right)$$

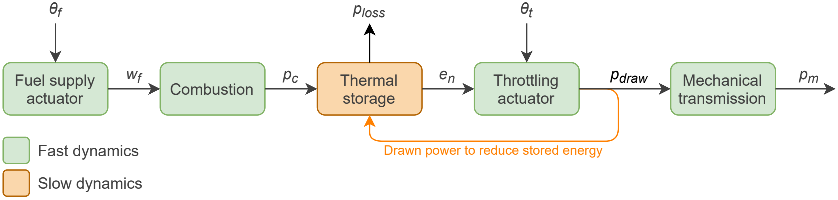
- which means, rearranging,

$$\delta\dot{\omega} = \frac{P_n}{J\omega_o^2}(\delta P_m - \delta P_e)$$

Note that $[P_m/J\omega_o^2] = [W/J] = [1/s]$; the quantity $J\omega_o^2/P_m$ is typically denoted by T_A.

Thermoelectric generators

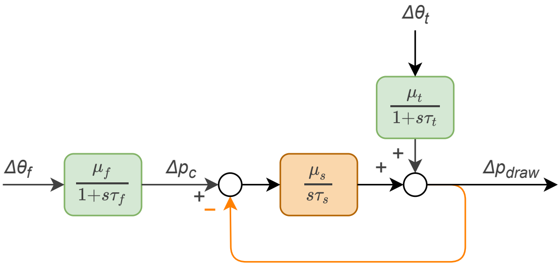
A system-level model – synthetic scheme for the prime mover



- From the system-level viewpoint, acting on the fuel actuator (θ_f) or on the throttling one (θ_t) influences power p_{draw} through a dynamics that can be *decently* represented with a 2nd order system;
- θ_t acts rapidly but its effect need sustaining in the long run by p_c;
- on the contrary θ_t produces a sustained effect on p_{draw} via p_c, but has to traverse the far slower thermal dynamics.

Thermoelectric generators

A system-level model – synthetic scheme for the prime mover – QUALITATIVE interpretation



- The effect of θ_t on draw as per the model is in fact multiplicative —yet additive (linearising) in the small.
- From the scheme above we get

$$G_{pf}(s)=\frac{\Delta p_{draw}(s)}{\Delta\theta_f(s)}=\frac{\mu_f}{(1+s\tau_f)\left(1+s\frac{\tau_s}{\mu_s}\right)},\quad G_{pt}(s)=\frac{\Delta p_{draw}(s)}{\Delta\theta_t(s)}=\frac{s\mu_t\frac{\tau_s}{\mu_s}}{(1+s\tau_f)\left(1+s\frac{\tau_s}{\mu_s}\right)},$$

where the f, t and s subscripts stand for fuel, throttling, storage.

Thermoelectric generators

A system-level model – synthetic scheme for the prime mover – QUALITATIVE interpretation

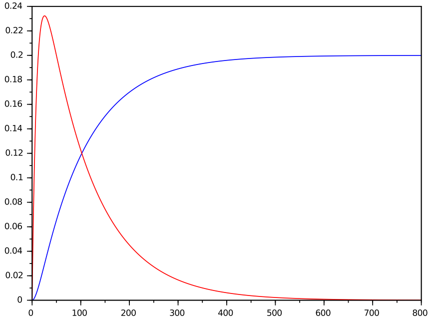
- Let us simulate the responses of G_{pf} and G_{pt} in Scilab with ballpark numbers, just to see their aspect:

```
muf  = 0.2;
tauf = 10;
mut  = 0.3;
taut = 1;
mus  = 1;
taus = 100;
Gpf  = syslin('c',muf/(1+%s*tauf)/(1+%s*taus/mus));
Gpt  = syslin('c',%s*mut*taus/mus/(1+%s*tauf)/(1+%s*taus/mus));
t    = 0:0.1:800;
yf   = csim('step',t,Gpf);
yt   = csim('step',t,Gpt);
plot(t,yf,'b',t,yt,'r');
```

Thermoelectric generators

A system-level model – synthetic scheme for the prime mover – QUALITATIVE interpretation

- Result:



- As can be seen, the representation is "decent" in the sense suggested;
- in real-life cases higher order models can be used, but the dominant dynamics do resemble those we just saw.

Thermoelectric generators

A system-level model – synthetic scheme for the prime mover – QUALITATIVE interpretation

- Remark: the zero in the origin in G_{pt} comes from neglecting losses.
- This is consistent with our previous viewpoint as per the transfer matrix $\Gamma(s)$, since

$$\lim_{T_{loss}\to\infty}\Gamma(s)=\frac{1}{1+s\frac{T_{draw}}{\overline{\theta}}}\begin{bmatrix}\frac{s\,\eta_m T_{draw}^2\overline{p}_c}{T_{rest}\overline{\theta}^2} & \frac{\eta_m T_{draw}}{T_{rest}} \\ -\frac{T_{draw}\overline{p}_c}{T_{rest}\overline{\theta}^2} & \frac{T_{draw}}{T_{rest}\overline{\theta}}\end{bmatrix}$$

where θ is playing the role now played by θ_t (if not for the cascaded dynamics of the throttling actuator).

- Note also that all the gains (for $\Gamma_{1,1}$, *lato sensu*) contain T_{draw}/T_{rest}, i.e., "how slow you restore over how slow you take".

Controlled (thermo) generators

Foreword

- We said that we would not represent the controls inside generators.
- True, but we now have to add a bit of additional information (still about the thermo case, before we generalise as anticipated).

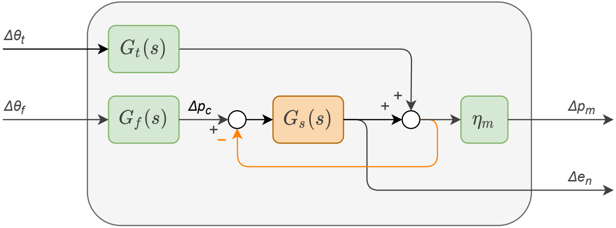
- After going through several coordinated viewpoints, we ended up with a (thermo) generator 2×2 linearised model
    - having as inputs the variations $\Delta\theta_f$ and $\Delta\theta_t$ of the fuel and throttling [0,1] commands,
    - and as outputs the variations Δp_m of the normalised mechanical power (i.e. for Δp_{draw}, as we assume η_m constant) and Δe_n of the normalised energy content (corresponding in practice to a normalised pressure).

- The generator internals are discussed in other courses,
- but here, to get to out final representation, we need to spend a few words on how θ_f and θ_t can be coordinated.

Controlled (thermo) generators

The 2×2 system to control

- Precisely, we got to



where the throttling, fuel/combustion and storage dynamics are

$$G_t(s) = \frac{\mu_t}{1+s\tau_t}, \quad G_f(s) = \frac{\mu_f}{1+s\tau_f}, \quad G_s(s) = \frac{\mu_s}{s\tau_s}.$$

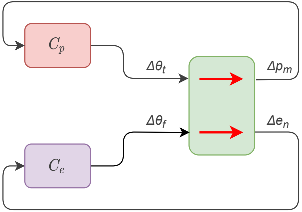
Controlled (thermo) generators

Alternatives for coordinating θ_t and θ_f

- We consider three alternatives:
  - using θ_t to control p_m and θ_f to control e_n, which is called boiler follows;
  - using θ_t to control e_n and θ_f to control p_m, which is called turbine follows;
  - setting the throttling valve to full open and use θ_f to control p_m, which is called variable or sliding pressure;

Controlled (thermo) generators

Boiler follows (BF)

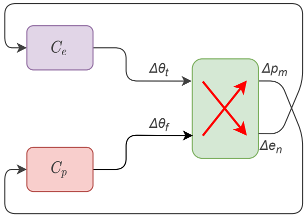


- Power control (C_p) via θ_t, energy control (C_e) via θ_f:
  - ☺ fast power response as τ_t is tendentiously small (seconds or below);
  - ☹ transient pressure (energy) variations of potentially noticeable entity, hence mechanical stress that can be detrimental in the long run;

  $\Rightarrow$ advisable when a plant has to take care of following fast load variations.

Controlled (thermo) generators

Turbine follows (TF)

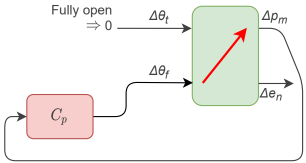


- Power control (C_p) via θ_f, energy control (C_e) via θ_t:
  - ☺ almost ideal pressure (energy) control, hence little mechanical stress;
  - ☹ slow power response, as θ_f acts through the thermal dynamics (although energy control helps by its fast action on pressure);

  $\Rightarrow$ advisable when a plant has to take care only of quite slow load variations.

Controlled (thermo) generators

Sliding pressure (SP)



- Power control (C_p) via θ_f, $\theta_t = 1$:
  - ☺ minimum stress for the turbine as there is no control action on the throttling valve;
  - ☹ extremely slow power response, as θ_f acts through the thermal dynamics but this time there is no energy (pressure) loop to help;

  $\Rightarrow$ advisable for base load plants (switching to BF/TF if pressure varies too much).

Controlled (thermo) generators

Conclusions

- Blocks C_p and C_c are generally quite simple, such as PI/PID ones.
- As a consequence, the relationship between power request and produced power (p_m) is reasonably represented – *from our system-level viewpoint* – with a transfer function of quite low order, say three at most, with possibly slightly underdamped dynamics.
- As such, in the rest of this course we shall represent a (thermo) generator as a single block
  - with a [0,1] input that we shall denote by u to stay abstracted with respect to the BF/TF/SP/combined internal control policy
  - and an output that we shall denote by P_g (for "generated") power as in general this can be of other nature than "mechanical".

Controlled (thermo) generators

Conclusions

- We shall however preserve a normalised power output, hence writing a generator block as
$$G(s) = \frac{\Delta P_g(s)}{\Delta u(s)} = P_n\, g(s)$$
where P_n is the nominal power and the transfer function $g(s)$, of unitary gain, outputs the normalised generated power $p_g = P_g/P_n$.
- Note: we have to always bear in mind that our transfer function models come from the linearisation of more complex ones not in the scope of this course, whence the Δ's; nonetheless, to lighten the notation, in the following we may sometimes drop the said Δ's (a simplification that must not be interpreted as altering in any sense the linearised and thus local nature of the used models, however).

Controlled (thermo) generators

Conclusions

- As for $g(s)$, we shall use
  - simple first- or overdamped second-order models such as

$$g(s) = \frac{1}{1+sT_g}, \quad g(s) = \frac{1}{(1+sT_{g1})(1+sT_{g2})}$$

    for exercises where the detailed form of $g(s)$ is not the point;
  - slightly more complex models like e.g.

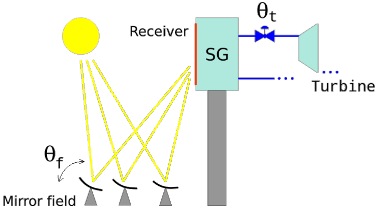
$$g(s) = \frac{1}{(1+sT_g)\left(1+2\frac{\xi_g}{\omega_{ng}}s+\frac{s^2}{\omega_{ng}^2}\right)}$$

    in some illustrative simulation examples.
- As we shall see, simple models like these can well reproduce the typical power/frequency transients of interest for network control and management.

Solar plant

of the thermal type – traditional setup

- Heliostats (mirrors) point to receiver on the steam generator (SG):

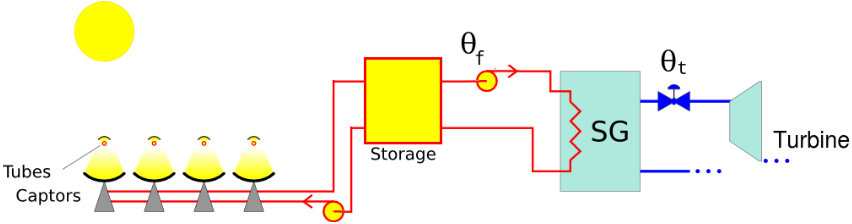


- The primary energy source (sun) is clearly uncontrollable.
- What is controllable, by focusing or de-focusing the mirrors, is the *amount* of the available power that the plant actually draws
  $\Rightarrow$ the mirror focusing plays more or less the role of θ_f, but subject to the variability above (representable as a variable μ_f).
- Main problems: the said variability and the difficulty of introducing "large" energy storages (w.r.t. that provided by the SG alone).

Solar plant

of the thermal (thermodynamic) type – alternative setup

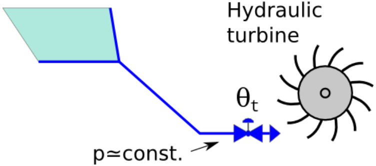
- An example:



- The primary fluid (e.g., molten salt) allows for a significant heat storage, thereby smoothing the source variability seen by the SG $\Rightarrow$ the situation is more similar to the reference thermo case.
- Note: the figure is *highly* simplified as for the hot fluid storage management.

Hydro plant

- Very simple scheme:



  - At the generator control time scale, the energy reserve (basin) may be considered unlimited. Also, the piping height typically dominates that of the basin ⇒ constant pressure at the valve inlet.
  - The mechanical power can be assumed to be an algebraic function of the turbine (throttling) valve command, i.e.,

$$P_g = f(\theta_t),$$

    although rigorously depending also on rotor and fluid speed (losses are due e.g. to residual jet kinetic energy).
- ...and so forth for other generator types.

Generators without rotating masses

(just a couple of words)

- Notable examples are photovoltaic generators and fuel cells.
- In both cases there is a primary source, either vastly exogenous (solar radiation) or well controllable (fuel).
- Then, there may or may not be a significant reserve (this is not the case e,g, for a photovoltaic generator without batteries).
- The process of generating electricity does not involve mechanics and is mostly solid-state, thus one can think that power can be commanded independently of frequency, while synchronisation with the network is always guaranteed.
- In one word, these generators as well fall in our system-level model structure. Enough on the matter for this course.

Transmission lines

in AC networks

- A transmission line is readily represented as the series of a resistance and an inductance.
- In general, the latter dominates.
- For problems requiring an explicit representation of voltages and currents (such as load flow) the natural modelling approach is in the phasor domain, as a complex (almost imaginary) impedance.
- For problems just requiring represent powers as signals, and in particular concentrating on active power as in frequency control, lines are in fact just losses, and can be represented (if necessary) by efficiencies conveniently located in the network.

Transmission lines

in AC networks

- As anticipated, for the type of problems we address, we are not representing transformers.
- The matter is treated in specialised courses, as is reactive power control and network contingency management.
- When necessary, for the problems we face, one could just account for the inevitable losses across transformers, resort to an efficiency-based description here as well.

Generic loads

- When representing voltages and currents, loads can be either
  - impedances (obviously)
  - or devices drawing a specified active (and reactive) power thanks to some internal controls, as is more and more frequently the case with inverters.
- When representing just power flows as signals, load are clerly just part of such signals, either exogenous or – when talking about "demand side control" – subjected to some management policy instead of just coming from requests on the part of the utilisers.
- We shall be more specific on the matter just sketched when it comes to control problems.

Electric systems: power and frequency control

Islanded generator case

Multiple generators (network) case

A few words on swinging

Preliminaries

- We start from a generator model in the known form

$$G(s) = \frac{\Delta P_g(s)}{\Delta u(s)} = P_n\, g(s)$$

  where P_n is the nominal power and $g(s)$ depends on the generator dynamics time scales, in turn tied to both size and management policy.
- We consistently take an energy-centred approach, i.e., there is no explicit evidence of voltages and currents, and powers are treated as signals.
- We assume the electric power demand P_e – precisely for our context, its *variation* ΔP_e – to be exogenous.
- We consider a single generator feeding its own load, which is named the islanded case.

Preliminaries

- We recall the balance at the alternator shaft, that replacing for generality the m (mechanical) subscript with the g (generated) one, reads

$$\delta\dot{\omega} = \frac{P_n}{J\omega_o^2}(\delta P_g - \delta P_e)$$

where lowercase δ's denote normalised variations, i.e.,

$$\delta\omega = \frac{\Delta\omega}{\omega_o} = \frac{\omega - \omega_o}{\omega_o}, \quad \delta P_{g,e} = \frac{\Delta P_{g,e}}{P_n} = \frac{P_{g,e} - P_n}{P_n};$$

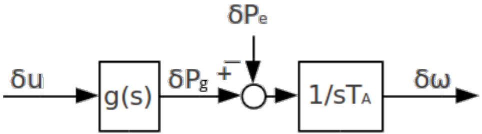
- notice that we normalise both P_g and P_e w.r.t. P_n.
- We finally recall that $J\omega_o^2/P_m$ has the dimensions of time, and is denoted by T_A; also, $u \in [0,1]$, hence $\delta u = \Delta u$.

Controlled system

- As a result, in the Laplace domain we can write

$$\delta\omega(s) = \frac{1}{sT_A}\left(\delta P_g(s) - \delta P_e(s)\right) = \frac{1}{sT_A}\left(g(s)u(s) - \delta P_e(s)\right),$$

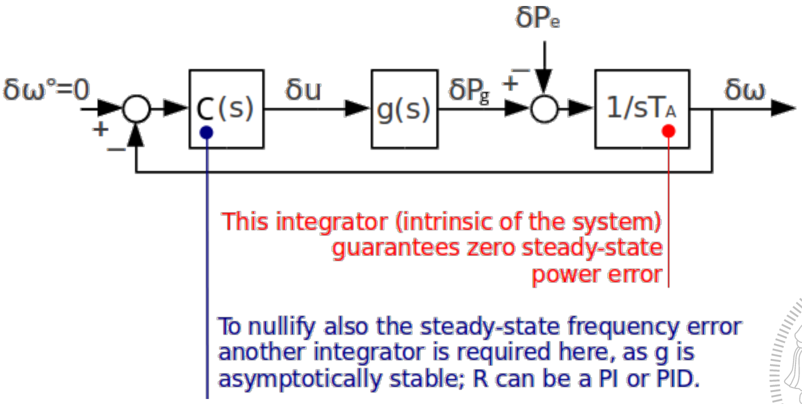
- and therefore in block diagram form



where the disturbance role of δP_e is evidenced.

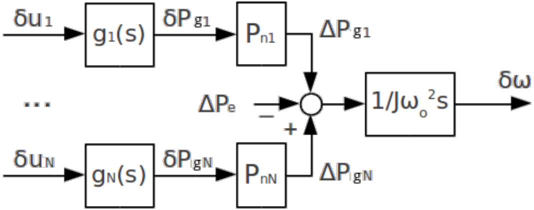
- Recall that $g(s)$ has low order, and above all is asymptotically stable.

Control scheme



Networked generators

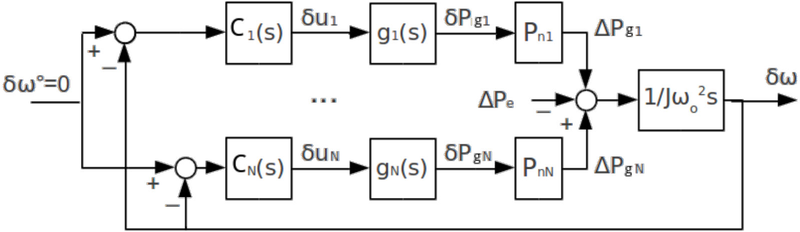
- In the case of multiple generators, at our system level we introduce the rigid synchronous network hypothesis: all the masses rotate together at the same speed, no swinging.
- In this case all the mechanical powers (*not the normalised powers*, beware) sum together, while there is still a *single* electric power demand (the total for the network) subtracted from them. The system under control is thus



where J is the total network inertia (there is no overall T_A as each generator has its own P_n).

Networked generators

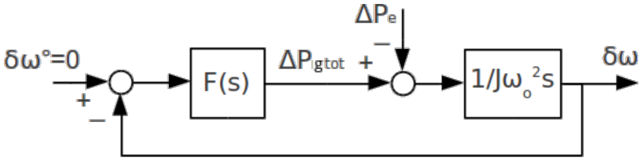
- The scheme for the islanded generator is easily extended to multiple generators as



- Here too the network intrinsic integrator $(1/J\omega_o^2 s)$ guarantees zero steady-state power error.
- However the regulators in this scheme cannot encompass integrators, because in that case the generation distribution would not be controllable.

Networked generators

- To understand why, observe that the scheme is equivalent to



- where ΔP_{mtot} is the total mechanical power variation (*not normalised*), and

$$F(s) = \sum_{i=1}^{N} R_i(s) g_i(s) P_{ni}.$$

- Possible integrators in the R_i regulators would thus be in parallel, whence the controllability loss.

Networked generators

- Solution: to have zero steady-state frequency error there must be one integrator.
- Therefore
  - we employ for the *primary regulators* $R_i(s)$ a type 0 structure (most frequently a pure proportional term K, whence the name "$K\Delta f$" frequently encountered for them),
  - and we introduce a *secondary frequency control* in the form of a single integrator per network, having as input the frequency error;
  - the output of the secondary regulator acts as an additive correction on the output of each $K\Delta f$ controller via a gain β_i that can be different for each generator, and dictates how much that generator will be asked to participate into secondary control.

Networked generators

Control scheme

SCHEME

Example 1

Islanded generator with (dominantly) first-order dynamics

Foreword

- It is sometimes useful to account for a non-rigid network.
- This means that generators can (transiently) rotate at different frequencies.
- To represent the involved phenomena, a mechanical equivalent of the generators' interconnection is the most adequate level of detail.
- To address the matter, we need however to first discuss how power is transferred in AC networks with controlled voltage.
- As usual (and with no didactic detriment) we assume a single voltage throughout the network, and a single-phase system.
- We also assume the network frequency to be controlled (now we know how), hence we can use phasors.
- We finally assume a prevailing network, i.e., that each generator is individually so small, compared to the union of the others, to see the network voltage as a fixed phasor.

Foreword

- In the above hypotheses one could see a contradiction between ideally controlled frequency, and generators accelerating/decelerating.
- What we precisely mean is that a generator can *transiently* accelerate or decelerate to change its voltage angle with respect to the network one (motivations follow),
- but then the generator recovers synchronisation with the network frequency,
- and the entity of such relative angle movements are individually associated to so small powers with respect to the total, to still allow to see the network voltage phasor as fixed.
- This said, we can proceed.

Power transfer

- Consider a generator connected to the network through an admittance $\underline{Y}_{gn}$.
- Let $\underline{V}_g$ and $\underline{V}_n$ be the generator and network voltage phasors.
- The current flowing from generator to network is therefore

$$\underline{I}_{gn} = \underline{Y}_{gn} \left(\underline{V}_g - \underline{V}_n \right),$$

- and the complex power leaving the generator is

$$\underline{S}_{gn} = \underline{V}_g \left(\underline{Y}_{gn} \left(\underline{V}_g - \underline{V}_n \right) \right)^*,$$

where, recall, * denotes the complex conjugate.

Power transfer

- Denoting by δ_{gn} the angle between phasors $\underline{V}_g$ and $\underline{V}_n$,some Maxima gives us the active and reactive power (P_g and Q_g, respectively) leaving the generator as

```
Vn  : Vnm;                          /* magnitude Vnm, phase 0 (reference)  */
Vg  : Vgm*(cos(dgn)+%i*sin(dgn));   /* magnitude Vgm, phase dgn            */
Ygn : Ggn-%i*Bgn;
Ign : (Vg-Vn)*Ygn;
Sgn : Vg*conjugate(Ign);
Pg  : trigsimp(realpart(Sgn));
Qg  : trigsimp(imagpart(Sgn));
```

- We get

$$\begin{aligned} P_g &= G_{gn}|\underline{V}_g|^2 - |\underline{V}_g||\underline{V}_n|\left(G_{gn}\cos\delta_{gn} - B_{gn}\sin\delta_{gn}\right) \\ Q_g &= B_{gn}|\underline{V}_g|^2 - |\underline{V}_g||\underline{V}_n|\left(B_{gn}\cos\delta_{gn} + G_{gn}\sin\delta_{gn}\right) \end{aligned}$$

- As can be seen, acting on $|\underline{V}_g|$ and δ_{gn} allows to control both active and reactive power.

Power transfer

- For our purpose let us however assume ideal voltage control, i.e., $|\underline{V}_g| = |\underline{V}_g| = V$; this reduces the previous equations for P_g and Q_g to

$$
\begin{array}{rcl}
P_g & = & V^2\left(G_{gn}(1-\cos\delta_{gn}) + B_{gn}\sin\delta_{gn}\right) \\
Q_g & = & V^2\left(B_{gn}(1-\cos\delta_{gn}) - G_{gn}\sin\delta_{gn}\right)
\end{array}
$$

- Assuming $B_{gn} \gg G_{gn}$, the active power transferred to the network is proportional to the sine of the angle between the generator and the network voltage.
- This is called the machine angle.

Power transfer

- We can easily check this with a bit of Maxima:

```
Vn  : V;
Vg  : V*cos(dgn)+%i*V*sin(dgn);
Ygn : Ggn-%i*Bgn;
Ign : (Vg-Vn)*Ygn;
Sgn : Vg*conjugate(Ign);
Pg  : factor(trigsimp(realpart(Sgn)));
Qg  : factor(trigsimp(imagpart(Sgn)));
```

A mechanical equivalent model

Rationale and interface

- We can represent the connection between generators and network by means of a Modelica connector carrying an angle as effort variable and a power as flow variable (similar to a mechanical rotational flange, just with power instead of torque for convenience).
- This is specified as

```
connector PowerAnglePort;
       Modelica.SIunits.Angle       theta;
  flow Modelica.SIunits.ActivePower P;
end PowerAnglePort;
```

- Notice the use of SI units through the Modelica Standard Library.

A mechanical equivalent model

Equations — we stick here to the m subscript for the generated power

$$
\begin{array}{rcll}
\theta_{port} & = & \text{port.theta} & \text{Connector equations for angle} \\
P_e & = & \text{port.P} & \text{and power} \\
 & & & \\
\omega_{port} & = & \dot{\theta}_{port} & \text{Angular velocities} \\
\omega_g & = & \dot{\theta}_g & \\
 & & & \\
J_g\dot{\omega}_g & = & \tau_m - \tau_e & \text{Momentum balance for generator inertia} \\
P_e & = & \omega_{port}\tau_e & \text{Power/torque/velocity equation, electrical} \\
P_m & = & \omega_g\tau_m & \text{Same, mechanical} \\
\delta_g & = & \theta_g - \theta_{port} & \text{Machine angle w.r.t. port (network)} \\
\tau_e & = & K\delta_g + F\dot{\delta}_g & \text{Equivalent spring/damper} \\
 & & & \\
P_m & = & G(s)u & \text{Mechanical power from generator dynamics}
\end{array}
$$

A mechanical equivalent model

Parametrisation

- Parameters (we omit `Modelica.SIunits.`) are given as

```
ApparentPower  Srated  "rated power";
Frequency      fnom    "nominal frequency";
Real           dnom    "angle to yield rated power at sync speed [deg]";
Time           Tox     "proper oscillation period";
Real           xi      "oscillation damping factor";
```

- and those in the model just written are computed as

```
// nominal frequency in r/s
      AngularVelocity           wnom  = 2*Modelica.Constants.pi*fnom;
// nominal machine angle in r
      Angle                     dnomr = dnom/180*Modelica.Constants.pi;
// Srated,dnom->elasticity
      RotationalSpringConstant  K     = Srated/wnom/(dnom/180*Modelica.Constants.pi);
// K,Tox->inertia
      Inertia                   J     = K/(2*Modelica.Constants.pi/Tox)^2;
// K,J,xi->friction
      RotationalDampingConstant F     = 2*xi*sqrt(J*K);
```

A mechanical equivalent model

Loads (just a few words here)

- As for loads, a constant-power one is just composed of an input P for the active power to draw, and of the equation

$$\mathrm{port.P} = P.$$

- A linear droop one with power variation limit is conversely written as

```
parameter Power      Pnom   "nominal (active) power";
parameter Frequency  fnom   "nominal frequency";
parameter Real       droop  "DP/Pnom = droop * Df/fnom";
parameter Real       beta   "max per-unit P variation wrt Pnom";
...
final parameter SI.AngularVelocity wnom = 2*Modelica.Constants.pi*fnom;
...
w    = der(port.theta);
Pact = port.P;
Pact = max((1-beta)*Pnom,min((1+beta)*Pnom,P+droop*(w-wnom)/wnom*Pnom));
```

Example 1

Two generators and four loads (two with droop)

Electric systems – generation optimisation

A system-level model

with to reference the thermo case

- For simplicity we identify here cost and fuel consumption (i.e., we do not include plant maintenance, personnel and so on).
- Combustion is not equally efficient at all *plant loads*, i.e. – looking at the *thermal* load – for all values of p_c.
- A *specific* consumption c_s ([kg of fuel per J], i.e., [kg/s of fuel per W]) is thus defined, which is typically a decreasing function of P_c in the admissible operation range, which normalised as done for p_c, in turn corresponds to an interval $(p_{c,min}, p_{c,max})$:
  - $p_{c,min}$ is the minimum "technical" load below which the generator cannot be operated, and may be something like 0.2–0.25,
  - while $p_{c,max}$ is the maximum "guaranteed" power for the generator to work safely, and can be slightly greater than the unity, say 1.05–1.10, to accommodate for transient "exceptional" power releases to the network.

A system-level model

with reference to the thermo case

- Given the above, the fuel mass flowrate w_f and p_c are related by

$$w_f = c_s(p_c P_n) p_c P_n,$$

which in our model we use to compute w_f while for this treatise considering p_c as the control input.
- In fact for control we consider the input to be θ_f and we assume this to be linearly related to p_c: as far as an estimate of $c_s(P)$ is available, however, the matter is no conceptual issue as the said estimated curve is invariantly smooth enough to be compensated for.

Generator cost models

- Most frequently, cost models are polynomial in the generated power and up to cubic. A typical form is

$$c_i(P_{gi}) = (k_{g1}P_{gi} + k_{g2}P_{gi}^2 + k_{g3}P_{gi}^3)k_F + k_{om0} + k_{om1}P_{gi}$$

where

c_i	$[€/h]$	is the cost rate,
P_{gi}	$[W]$	is the generated power,
$k_{gj},\ j = 1\ldots3$	$[J/(hW^j)]$	are cost coefficients for pure generation,
k_F	$[€/J]$	is the fuel cost per unit of *energy*,
$k_{om\ell},\ \ell = 0,1$	$[€/(hW^\ell)]$	are cost coefficients for Operation & Maintenance.

- Note that, quite logically, the pure generation cost vanishes for $P_{gi} = 0$, while the O&M cost does not.

Generator cost models

- Why cubic?
- Because the fuel to generated power ratio $r_{fP}(P_g) = Q_f(P_g)/P_g$, where Q_f is the power yielded by fuel $[W]$ and P_g the generated power, thus making r_{fP} adimensional, typically has a minimum at the *optimal operating point* (by construction generally close to the maximum or *rated* power $P_{g,max}$).
- A good way to synthetically model this is to describe function $r_{fP}(P_g)$ as a *parabola*, specifying
  - the optimal (minimum) fuel to generated power ratio r^o_{fP},
  - the fraction p^o_g of $P_{g,max}$ corresponding to that optimal ratio, where p_g is defined as $P_g/P_{g,max}$,
  - and the fuel to generated power ratio r^{ml}_{fP} ($> r^o_{fP}$) at the minimum sustainable load, i.e., at $p^{ml}_g = P_{g,min}/P_{g,max}$,

Generator cost models

- The above computations give

$$r_{fP}(p_g) = r_{fP}^o + \frac{r_{fP}^{ml} - r_{fP}^o}{\left(p_g^o - p_g^{ml}\right)^2}\left(p_g - p_g^o\right)^2$$

thus

$$r_{fP}(P_g) = r_{fP}^o + \frac{r_{fP}^{ml} - r_{fP}^o}{\left(P_g^o - P_{g,min}\right)^2}\left(P_g - P_g^o\right)^2$$

Generator cost models

- Alternatively, $r_{fP}(P_g)$ can be obtained by interpolating experimental points (and again, a *parabola* normally suffices).
- Therefore, no matter how $r_{fP}(P_g)$ is obtained, $Q_f(P_g)$ can be expressed as

$$Q_f(P_g) = r_{fP}(P_g)P_g = \left(r_{fP}^o + \frac{r_{fP}^{ml} - r_{fP}^o}{\left(P_g^o - P_{g,min}\right)^2} \left(P_g - P_g^o\right)^2 \right) P_g$$

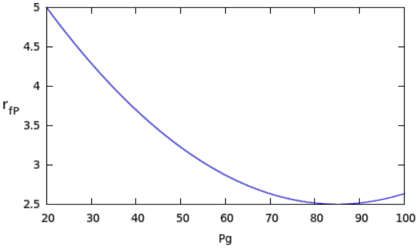
  that apparently contains powers of P_g from one to three, like we just wrote for the term $k_{g1}P_{gi} + k_{g2}P_{gi}^2 + k_{g3}P_{gi}^3$.
- In detail,

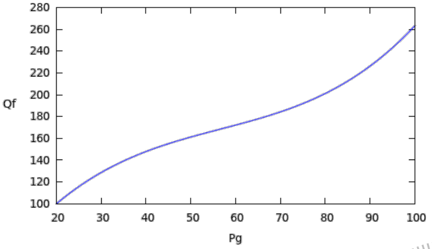
$$k_{g1} = \frac{r_{fP}^{ml} P_g^{o\,2} - r_{fP}^o P_{g,min}\left(2P_g^o - P_{g,min}\right)}{\left(P_g^o - P_{g,min}\right)^2}, \quad k_{g2} = \frac{2P_g^o \left(r_{fP}^o - r_{fP}^{ml}\right)}{\left(P_g^o - P_{g,min}\right)^2},$$
$$k_{g3} = \frac{r_{fP}^{ml} - r_{fP}^o}{\left(P_g^o - P_{g,min}\right)^2}.$$

Generator cost models

An example

- The above cost model with $r_{fP}^o = 2.5$, $r_{fP}^{ml} = 5$, $P_g^o = 85$, $P_{g,min} = 20$ (supposing $P_{g,max} = 100$ for the plots) produces





and, for completeness,

$$k_{g1} = 6.775, \quad k_{g2} = -0.101, \quad k_{g3} = 5.917 \cdot 10^{-4}.$$

- Note: r_{fP} can be interpreted as the *inverse* of the fuel-to-power *efficiency* $\eta_{fP}(P_g) = P_g/Q_f(P_g)$, ranging in this case from 0.4 (optimal point) to 0.2 (minimum sustainable load).

Problem statement

Preliminaries

- To understand, better *not* to reason with normalised quantities.
- Consider a network with N generators: at any given moment, the total generated power must equal the electric power demand, i.e.,

$$\sum_{i=1}^{N} P_{gi} = P_e$$

- Primary and secondary control can ensure this, and also keep frequency to the set point (if the power request is feasible, of course).
- But as said, what about cost?
- Knowing the efficiency curve of each generator, one can write N functions to relate each $P_{gi}\,[W]$ to a "cost rate" $c_i\,[€/s]$ or $[€/h]$.
- Of course each $c_i(P_{gi})$ is a monotonically increasing function and each generator has limits, i.e., $P_{gi,min} \leq P_{gi} \leq P_{gi,max}$.

Problem statement

Cost function

- Supposing for this first case that the purpose is to minimise the overall cost, it can be stated as that of minimising the overall cost rate, hence as

$$
\begin{array}{ll}
\min \sum_{i=1}^{N} c_i(P_{gi}) & \\
\text{s.t.} & \sum_{i=1}^{N} P_{gi} = P_e, \\
 & P_{gi,min} \leq P_{gi} \leq P_{gi,max},\ i = 1 \ldots N.
\end{array}
$$

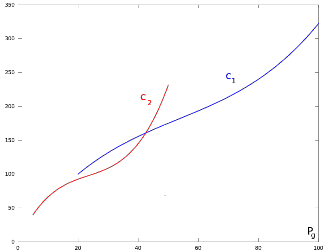
- Note: the problem can be much more complex as *sets* of generators may aim for example at minimising *their* cost while together generating a stipulated total power or a given share of P_e, or at maximising their revenue irrespectively of the rest, and so forth.
- We shall face some additional complexity later on, compatibly with the available space.
- For the moment, let us understand the principles.

An introductory case

Two generators

- Generator data:

$$
\begin{array}{lllll}
P_{g1,max} = 100 & P_{g1,min} = 20 & P^o_{g1} = 80 & r^o_{fP1} = 3 & r^{ml}_{fP1} = 5 \\
P_{g2,max} = 50 & P_{g2,min} = 5 & P^o_{g2} = 35 & r^o_{fP2} = 3.5 & r^{ml}_{fP2} = 8
\end{array}
$$



- Network power demand: $P_{e,max} = 130$, $P_{e,min} = 10$.
- We suppose that generators can be activated/deactivated at any time (this impacts operation scheduling, not in our scope).

An introductory case

Two generators

- Basic idea (which is general w.r.t. the example):
  - take the forecast power demand $\widehat{P}_e$ for the next "period" (day, hour,...),
  - determine the optimal generation distribution $\{P_{gi}^{opt}\}$ yielding $\widehat{P}_e$ at minimum cost,
  - send each of the so obtained generation requests P_{gi}^{opt} to the corresponding generator as a bias value,
  - and let primary and secondary control act as usual.
- Let us now concentrate on the $\widehat{P}_e \mapsto \{P_{gi}^{opt}\}$ problem, other aspects (to the feasible extent) later on.

An introductory case

Two generators

- Consider all the generator combinations, and determine the minimum and maximum power that can be generated by each of them:

$$G1 \to [20,100], \quad G2 \to [5,50], \quad G1+G2 \to [25,150].$$

- Consequently, divide the $\widehat{P}_e$ range in intervals I_i and determine the feasible combinations for each of the said intervals (easier to show than to explain):

```
                20                              100
G1              |--------------------------------|
          5                 50
G2        |-----------------|
                  25                                                  150
G1+G2             |--------------------------------------------------|
```

I_1	$[10,20]$	$\to$	G2
I_2	$[20,25]$	$\to$	G1,G2
I_3	$[25,50]$	$\to$	G1,G2,G1+G2
I_4	$[50,100]$	$\to$	G1,G1+G2
I_5	$[100,130]$	$\to$	G1+G2

An introductory case

Two generators

- For combinations with more than one generator, the optimal distribution (minimum total cost) has to be found.
- With only two generators the only case to consider is G1+G2, and we can proceed by substitution (we shall see something more general later on):

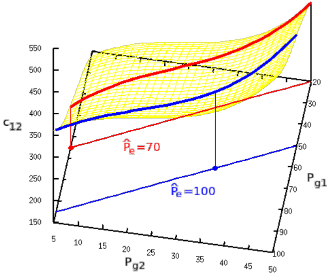
$$P_{g2} = \widehat{P}_e - P_{g1} \Rightarrow c_{12}(P_{g1}) = c_1(P_{g1}) + c_2(\widehat{P}_e - P_{g1})$$

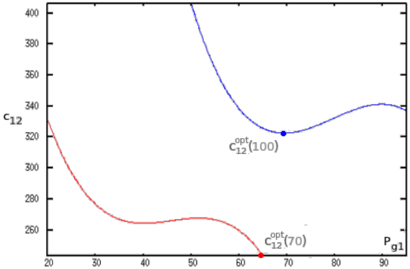
- Then we take the first and second derivative of $c_{12}(P_{g1}$ w.r.t. the only remaining independent variable P_{g1},
- find a possible minimum cost $c_{12}^{opt}(\widehat{P}_e)$ inside the power range of both generators (otherwise the minimum is at one of the two distribution extrema),
- determine the power distribution – i.e., $P_{g1}^{opt}(\widehat{P}_e)$ – corresponding to the minimum.

An introductory case

Two generators

- Graphical interpretation for the G1+G2 combination:



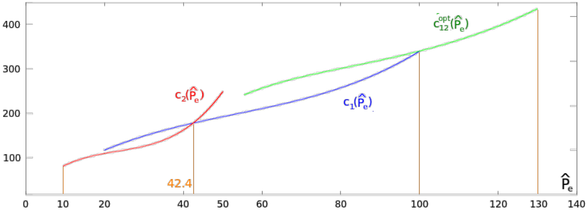


- As can be seen, for a given $\widehat{P}_e$, the optimal distribution can be inside the segment of the $P_{g1} + P_{g2} = \widehat{P}_e$ straight line, or at one of its extrema.

An introductory case

Two generators

- As illustrated by the previous graphical interpretation, for each value of $\widehat{P}_e$, the combination can now be chosen that provides the minimum cost; this is shown below:



- Finally, based on the choice just made, the bias (or "tertiary control") values $P_{b1,2}$ for $P_{g1,2}$ are determined.
- Let us now quit this introductory example and naïve technique, and move toward establishing a methodology.
- Before, however, we need to review some mathematics.

Optimisation background

Constrained optimisation – Lagrange multipliers – Karush-Kuhn-Tucker (KKT) equations

Problem statement

- We want to minimise a real function f of N_x real variables x_i, i.e.,

$$f(x_1, x_2, \ldots x_{N_x}), \qquad f(\cdot, \cdot, \ldots \cdot) \in \Re,\ \ x_i \in \Re,\ i = 1 \ldots N_x,$$

  subject to N_e equality constraints in the form

$$g_i(x_1, x_2, \ldots x_{N_x}) = 0, \qquad g_i(\cdot, \cdot, \ldots \cdot) \in \Re,\ i = 1 \ldots N_e,$$

  and to N_i inequality constraints in the form

$$h_i(x_1, x_2, \ldots x_{N_x}) \geq 0, \qquad h_i(\cdot, \cdot, \ldots \cdot) \in \Re,\ i = 1 \ldots N_i.$$

- *Caveat:* this is *not* a math lecture. We shall assume that "everything is regular enough", and not even mention several hypotheses that would be necessary for a rigorous treatise.

Optimisation background

Only equality constraints – Lagrange multipliers

- Form the problem's *Lagrangian* as

$$L = f(x_1, x_2, \ldots x_{N_x}) + \sum_{i=1}^{N_e} \lambda_i g_i(x_1, x_2, \ldots x_{N_x})$$

introducing N_e additional real unknowns λ_i, named the *Lagrange multipliers*.

- Compute the gradients of L w.r.t. vectors $x = [x_1 \ldots x_{N_x}]' \in \Re^{N_x}$ and $\lambda = [\lambda_1 \ldots \lambda_{N_e}]' \in \Re^{N_e}$, i.e.,

$$\nabla_x L(x,\lambda) = \left[\frac{\partial L}{\partial x_1}\,\frac{\partial L}{\partial x_2}\ldots\frac{\partial L}{\partial x_{N_x}}\right], \qquad \nabla_\lambda L(x,\lambda) = \left[\frac{\partial L}{\partial \lambda_1}\,\frac{\partial L}{\partial \lambda_2}\ldots\frac{\partial L}{\partial \lambda_{N_e}}\right],$$

having respectively N_x and N_e (function) components.

Optimisation background

Only equality constraints – Lagrange multipliers

- Observe that the k-th component of $\nabla_x L$ is

$$\frac{\partial L}{\partial x_k} = \frac{\partial f}{\partial x_k} + \sum_{i=1}^{N_e} \lambda_i \frac{\partial g_i}{\partial x_k} = \frac{\partial f}{\partial x_k} + \lambda' \cdot \begin{bmatrix} \frac{\partial g_1}{\partial x_k} \\ \vdots \\ \frac{\partial g_{N_e}}{\partial x_k} \end{bmatrix}$$

  where $\cdot$ denotes the scalar product. Therefore

$$\nabla_x L = \nabla_x f + \lambda' \cdot \begin{bmatrix} \frac{\partial g_1}{\partial x_1} & \cdots & \frac{\partial g_1}{\partial x_{N_x}} \\ \vdots & & \vdots \\ \frac{\partial g_{N_e}}{\partial x_1} & \cdots & \frac{\partial g_{N_e}}{\partial x_{N_x}} \end{bmatrix} = \nabla_x f + \lambda' \cdot \begin{bmatrix} \nabla_x g_1 \\ \vdots \\ \nabla_x g_{N_e} \end{bmatrix} = \nabla_x f + \lambda' J_x g$$

  where $J_x g$ is the Jacobian of the constraints g w.r.t. x.
- Also, observe that the k-th component of $\nabla_\lambda L$ is g_k.

Optimisation background

Only equality constraints – Lagrange multipliers

- Now, suppose that (x^o, λ^o) is a solution for the system of $N_x + N_e$ equations

$$\begin{cases} \nabla_x L(x,\lambda) & = & 0_{1\times N_x} \\ \nabla_\lambda L(x,\lambda) & = & 0_{1\times N_e} \end{cases}$$

  in the $N_x + N_e$ unknowns (x, λ), termed the Lagrangian Multipliers (LM) equations.
- The second equation says that x^o fulfils $g(x)$. For the first we have two cases
- Case 1: $\nabla_x f$ in x^o is a zero vector.
  - In this case x^o is a *stationary point* for $f(x)$ *independently of the constraints* $g(x)$.
  - In addition, given the expression of $\nabla_x L$, the equation reduces to $\lambda' J_x g = 0$; since (x^o, λ^o) fulfils it, either λ^o is a zero vector, or the gradients $\nabla_x g_i$, $i = 1 \dots N_e$ evaluated in x^o are linearly dependent.
  - If the $\nabla_x g_i$ are linearly dependent in x^o we shall then say that x^o *may* not be a *regular point* for the constraints g; we do not further discuss this matter.

Optimisation background

Only equality constraints – Lagrange multipliers

- Case 2: $\nabla_x f$ in x^o is not a zero vector.
  - In this case λ^o cannot be a zero vector either, or the considered equation $\nabla_x f + \lambda' J_x g = 0$ cannot be satisfied (contrary to the hypothesis).
  - Also, rewritten as $\nabla_x f = -\lambda' J_x g$, the same equation says that the gradients of $f(x)$ and $g(x)$ w.r.t. x are parallel in x^o.
  - Let now $z^o = f(x^o)$, and consider the hypercurve in $\Re^{N_x+1}$ obtained by intersecting the hypersurfaces $z = f(x)$ and $g(x) = 0$. Moving on that hypercurve away from (x^o, z^o), that apparently belongs to it, *locally* produces no variation of z. Therefore, x^o is a (local) stationary point for $f(x)$ constraine by $g(x) = 0$.
  - Since this may be hard to grasp, let us see an example with Maxima:

```
f : x1^2+x2^2;
g : x1-1;                              /* grad_x g = [1 0]          */
L : f+lam*g;
solve([diff(L,x1),diff(L,x2),
       diff(L,lam)],[x1,x2,lam]);
plot3d([f,g,0, [x1,0,2], [x2,-1,1]]);
subst([x1=1,x2=0],jacobian([f],[x1,x2])); /* grad_x f || grad_x g in xo */
subst([x1=1,x2=1],jacobian([f],[x1,x2])); /* and not e.g. here          */
```

Optimisation background

Only equality constraints – Lagrange multipliers

v

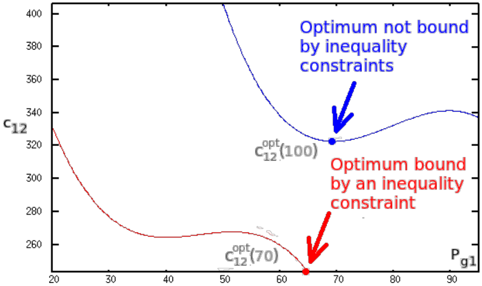
- Conclusion: x^o fulfils the LM equations
  $\Rightarrow$ it is a *candidate* constrained optimal point.

- We have then found a set of *necessary*, *first-order* conditions for *local* constrained optimality.
- Further studies could e.g. study via Hessian analysis whether is x^o a minimum, a maximum, or neither.
- For our purposes we can stop here, however we need to address *inequality* constraints.

Optimisation background

Equality and inequality constraints – Karush-Kuhn-Tucker (KKT) equations

- Major difference w.r.t. the equality-only (LM) case: a solution may *not* be a stationary point, as some inequality constraints may bind it.
- Observe that we have already found such a case in the introductory example with two generators:



- Hence the LM equations cannot be used here (i.e., the Lagrange *rationale* still works, but we need to introduce some modifications).

Optimisation background

Equality and inequality constraints – Karush-Kuhn-Tucker (KKT) equations

- We form again the Lagrangian, this time however in the form

$$\begin{aligned} L(x,\lambda,\mu) &= f(x)+\sum_{i=1}^{N_e}\lambda_i g_i(x)+\sum_{j=1}^{N_i}\mu_j h_j(x) \\ &= f(x)+\lambda\cdot g(x)+\mu\cdot h(x) \end{aligned}$$

  where another multiplier vector $\mu=[\mu_1\ldots\mu_{N_i}]'\in\Re^{N_i}$ is introduced w.r.t. the case with equalities only, and $h\in\Re^{N_i}$ is the vector of functions h_j,
- and give the following simple definitions:
  - if at a certain point x^o a certain inequality constraint h_j is satisfied with equality – i.e., if $h_j(x^o)=0$ – we shall say that the constraint h_j is *active* (or *binding*) in x^o;
  - if the constraint is satisfied with the $>$ sign – i.e., if $h_j(x^o)>0$ – we shall say that it is *inactive* (or *nonbinding*) in x^o;
  - otherwise (obviously) the constraint is *violated* in x^o.

Optimisation background

Equality and inequality constraints – Karush-Kuhn-Tucker (KKT) equations

- Now (we again look for necessary conditions) suppose that x^o is a solution (i.e., an optimal point) with neither binding nor violated inequality constraint, i.e., that $h_j(x^o) > 0 \forall j$.
- In this case x^o is also a solution for the LM problem, as setting $\mu = 0$ makes the term $\mu \cdot h(x^o)$ contribute zero to L.
- Note that also a solution for the LM problem *violating* some inequality constraint would fall in the same case, but it is not difficult to see if said constraints are violated or nonbinding. In the following we assume that such a *feasibility check* is always performed.
- The most interesting case is when at least one inequality constraint is binding. Let us expand a bit on this.

Optimisation background

Equality and inequality constraints – Karush-Kuhn-Tucker (KKT) equations

- By adopting the same notation introduced in the LM problem, consider the system of $N_x + N_e + N_i$ equations

$$\left\{\begin{array}{lclcl} \nabla_x L(x,\lambda,\mu) & = & \nabla_x f(x) + \lambda' J_x g(x) + \mu' J_x h(x) & = & 0 \\ \nabla_\lambda L(x,\lambda,\mu) & = & g(x) & = & 0 \\ \mu' \circ \nabla_\mu L(x,\lambda,\mu) & = & \mu' \circ h(x) & = & 0 \end{array}\right.$$

  in the $N_x + N_e + N_i$ unknowns (x,λ,μ), where $\circ$ denotes the Schur (element by element) product.

- Roughly speaking, a solution bound by some inequality constraints will satisfy an LM problem where the said constraints are fictitiously treated as equality ones (whence the last term in the first equation) provided that only those (binding) constraints are actually accounted for, which is ensured by the third equation (of course s.t. the necessary feasibility checks).

Optimisation background

Equality and inequality constraints – Karush-Kuhn-Tucker (KKT) equations

- The system above is composed of the so-called KKT equations.
- We observe that if a constraint $h_i(x) \geq 0$ is inactive *at optimality*, i.e. in x^o, then the corresponding μ_i zero.
- In the opposite case, the sign of μ_i dictates whether f increases or decreases when entering or exiting the admissible region as dictated by the inequality constraints.
- Assuming that we want to *minimise* $f(x)$ and the sign in the $h(x)$ equality constraints is $\geq$, requiring that $f(x)$ *increase* when x *enters* the feasibility region for all binding h_i – i.e., that a KKT solution with at least one binding inequality constraint be a candidate *bound minimum* – corresponds to requiring that all nonzero μ_i be *negative* in it. Of course all the other combinations are possible (we may want to maximise $f(x)$ and inequality constraints may have the $\leq$ sign).
- Let us now go for an elementary example.

Optimisation background

Equality and inequality constraints – KKT equations – elementary example

- Minimise $f(x) = x^2$ s.t. $x \geq 1$ $(\Rightarrow x-1 \geq 0)$, $x \leq 2$ $(\Rightarrow 2-x \geq 0)$.
- Maxima;

```
f       : x^2;
h1      : x-1;
h2      : 2-x;
L       : f+mu1*h1+mu2*h2;
KKTeqs  : [diff(L,x),mu1*diff(L,mu1),mu2*diff(L,mu2)];
solve(KKTeqs,[x,mu1,mu2]);
```

Optimisation background

Equality and inequality constraints – KKT equations – elementary example

- Solutions found:

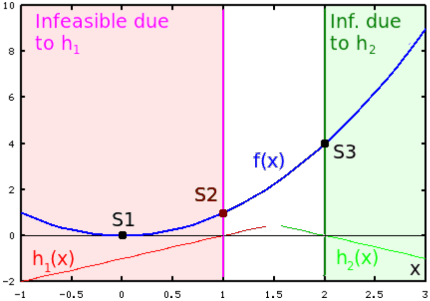
	x	μ_1	μ_2	$f(x)$	$h_1(x)$	$h_2(x)$
S1	0	0	0	0	-1	2
S2	1	-2	0	1	0	1
S3	2	0	4	4	1	0

- S1: infeasible.
- S2: feasible, h_2 nonbinding, h_1 binding and entering the feasible region increases f $\Rightarrow$ can be a bound minimum.
- S3: feasible, h_1 nonbinding, h_2 binding and entering the feasible region decreases f $\Rightarrow$ cannot be a bound minimum.
- Hence, the solution is S2.

Optimisation background

Equality and inequality constraints – KKT equations – elementary example

- This was *really* elementary, so let us have a visual look:



- S1: infeasible.
- S2: feasible, h_2 nonbinding, h_1 binding and entering the feasible region increases $f \Rightarrow$ can be a bound minimum.
- S3: feasible, h_1 nonbinding, h_2 binding and entering the feasible region decreases $f \Rightarrow$ cannot be a bound minimum.

Global cost minimisation

The previous two-generator case revisited with the KKT equations

- We are of course looking at the G1+G2 combination, thus the problem is
  minimise $f(P_{g1},P_{g2}) = c_1(P_{g1}) + c_2(P_{g2})$
  s.t. $P_{g1} + P_{g2} = \widehat{P}_e$, $P_{g1,min} \leq P_{g1} \leq P_{g1,max}$, $P_{g2,min} \leq P_{g2} \leq P_{g2,max}$.
- Maxima (preliminaries):

```
Pg1min : 20;
Pg1max : 100;
Pgo1   : 80;
ro1    : 3;
rml1   : 5;

Pg2min : 5;
Pg2max : 50;
Pgo2   : 35;
ro2    : 3.5;
rml2   : 8;

k11    : (rml1*Pgo1^2-ro1*Pg1min*(2*Pgo1-Pg1min))/(Pgo1-Pg1min)^2;
k21    : 2*Pgo1*(ro1-rml1)/(Pgo1-Pg1min)^2;
k31    : (rml1-ro1)/(Pgo1-Pg1min)^2;
k12    : (rml2*Pgo2^2-ro2*Pg2min*(2*Pgo2-Pg2min))/(Pgo2-Pg2min)^2;
k22    : 2*Pgo2*(ro2-rml2)/(Pgo2-Pg2min)^2;
k32    : (rml2-ro2)/(Pgo2-Pg2min)^2;

c1     : k11*Pg1+k21*Pg1^2+k31*Pg1^3;
c2     : k12*Pg2+k22*Pg2^2+k32*Pg2^3;
f      : c1+c2;
```

Global cost minimisation

The previous two-generator case revisited with the KKT equations

- Maxima (we consider the two cases with $\widehat{P}_e$ equal to 70 and 100):

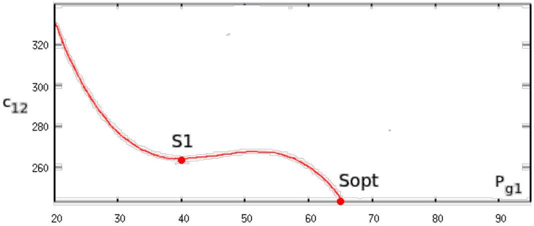
```
/* Set Pe to the desired value, 70 or 100 for the two cases shown */
Pe     : 70;

g      : Pg1+Pg2-Pe;
h1     : Pg1-Pg1min;
h2     : Pg1max-Pg1;
h3     : Pg2-Pg2min;
h4     : Pg2max-Pg2;

L      : f+lambda*g+mu1*h1+mu2*h2+mu3*h3+mu4*h4;
KKTeqs : [diff(L,Pg1),diff(L,Pg2),
          diff(L,lambda),
          mu1*diff(L,mu1),mu2*diff(L,mu2),
          mu3*diff(L,mu3),mu4*diff(L,mu4)];
S      :  solve(KKTeqs,[Pg1,Pg2,lambda,mu1,mu2,mu3,mu4]);

for i:1 thru length(%rnum_list) do S:subst(t[i],%rnum_list[i],S);
float(S);
fvals  : float(makelist(subst(S[i],f),i,1,length(S)));
```

Global cost minimisation

The previous two-generator case revisited with the KKT equations

- Results for $\widehat{P}_e = 70$:

P_{g1}	P_{g2}	λ	μ_1	μ_2	μ_3	μ_4	f	
20.00	50.00	any	$-\lambda-\frac{11}{3}$	0.00	0.00	$\lambda+\frac{97}{8}$	331.25	left extremum
51.58	18.42	-1.82	0.00	0.00	0.00	0.00	267.67	local max
40.09	29.91	-2.11	0.00	0.00	0.00	0.00	264.30	S1
100.00	-30.00	-44.12	0.00	-38.68	0.00	0.00	-416.52	infeasible
65.00	5.00	-2.04	0.00	0.00	-4.46	0.00	243.13	Sopt



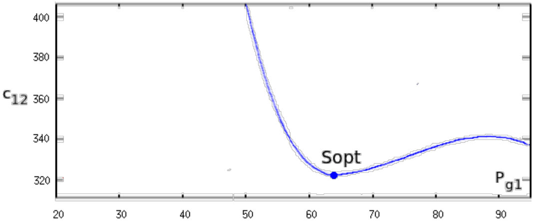
- Note that the left extremum cannot be a bound minimum, while the right one actually is.

Global cost minimisation

The previous two-generator case revisited with the KKT equations

- Results for $\widehat{P}_e = 100$:

P_{g1}	P_{g2}	λ	μ_1	μ_2	μ_3	μ_4	f	
89.75	10.25	-4.02	0.00	0.00	0.00	0.00	341.26	local max
69.42	30.58	-2.25	0.00	0.00	0.00	0.00	322.59	Sopt
20.00	80.00	-49.63	45.95	0.00	0.00	0.00	1190.00	infeasible
100.00	0.00	-9.63	0.00	-4.18	0.00	0.00	322.22	infeasible
95.00	5.00	-4.71	0.00	0.00	-1.79	0.00	336.88	right extremum
50.00	50.00	-1.83	0.00	0.00	0.00	10.29	406.25	left extremum



- Note that the left extremum cannot be a bound minimum, while the right one could but is not.
- Recall that the KKT equations not only are necessary conditions, but just provide *candidate optima*: always check the solutions!

Preliminaries

The generator viewpoint

- Assume you run a generator (or pool of) and that you are a price taker.
- Let π_σ – say [€/MWh] – be the energy selling price.
- Let $c(P_g)$ [€/h] be your cost rate as function of your generated power P_g [MW].
- To maximise your revenue (rate) you state the problem

$$\max_{P_g} \left(\pi_\sigma P_g - c(P_g) \right)$$

and readily get the production you desire by solving for P_g the equation

$$\frac{dc(P_g)}{dP_g} = \pi_\sigma .$$

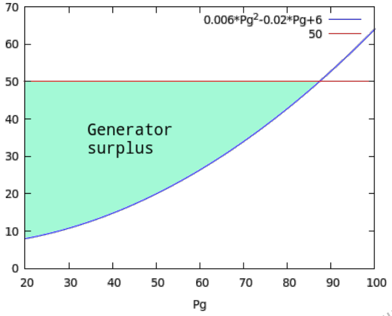
Preliminaries

The generator viewpoint – example

- Maxima:

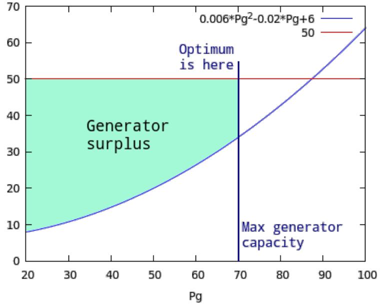
```
c   : 6*Pg-0.01*Pg^2+0.002*Pg^3;
sol : solve(diff(c,Pg)=psigma,Pg);
Pgo : rhs(sol[2]);
      float(subst(psigma=50,Pgo));
      plot2d([diff(c,Pg),50],[Pg,20,100]);
```

- The obtained optimum P_g is 87.32.
- Let us now plot $dc(P_g)/dP_g$ versus P_g and check the crossing with π_σ.
- NOTE: should the optimum P_g be above the generator capacity, the desired generation and the surplus would be bound to that maximum capacity.

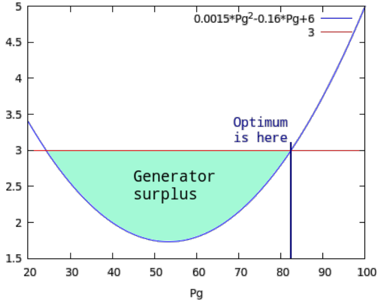


Preliminaries

The generator viewpoint – peculiar cases



Bound optimum.



High-inflection cost rate curve.

Preliminaries

The utiliser viewpoint

- Assume now you are an energy utiliser — and again, a price taker.
- Let π_σ – say again [€/MWh] – be the energy selling price.
- Assume your utility function $U(P_t)$ – [€/h], where P_t is the taken power – to be "well-behaved" in economic jargon, that is, monotonically increasing and (quasi-)concave; a typical form can be cubic without constant term and with negative second derivative in the range of interest.
- To maximise your utility (rate) you state the problem

$$\max_{P_t}\left(U(P_t)-\pi_\sigma P_g\right)$$

and get your optimal consumption by solving for P_t the equation

$$\frac{dU(P_t)}{dP_t}=\pi_\sigma.$$

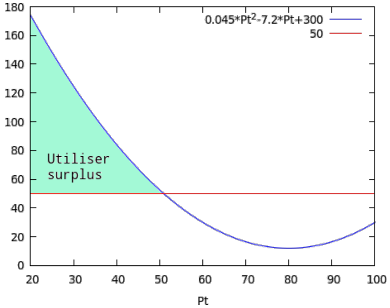
Preliminaries

The utiliser viewpoint – example

- Maxima:

```
U   : 300*Pt-3.6*Pt^2+0.015*Pt^3;
sol : solve(diff(U,Pt)=psigma,Pt);
Pto : rhs(sol[1]);
      float(subst(psigma=50,Ptopt));
      plot2d([diff(U,Pt),50],[Pt,20,100]);
```

- The obtained optimum P_t is 50.94.
- Let us now plot $dU(P_t)/dP_t$ versus P_t and check the crossing with π_σ.
- NOTE: should the optimum P_g be above the utilser capacity, the desired utilisation and the surplus would be bound to that maximum capacity.



Relating the two viewpoints

- Given a time slot in the future, generators and utilisers can bid for a certain amount of produced/consumed energy at a certain price in that slot. In a nutshell, this is how energy is traded.
- The matter is extremely complicated, in fact, involving auctions at various space and time scales.
- However this is a course on automation, hence we are not interested in the details of the bidding process – which is more a matter of economic management – but rather on how this process reflects on the control layers for the network.
- As such, we are now going for a wrap-up from our perspective.

Ingredients of the optimisation problem

- Every generator has a cost model providing the cost rate [€/h] as a function of the produced power [MW].
- Such cost (rate) functions are well represented, in general, by polynomials up to cubic.
- Utilisers of a certain size have – more or less explicitly – a utility function relating some KPI (for example an income rate) to the consumed power.
- Apparently the last idea makes little sense for a single house...
- ...but it could, no matter the KPI, e.g. for a district or a municipality.
- Such utility (rate) functions are, or at least in general are assumed to be, “well-behaved” (monotonic, increasing, concave).

Actors of the optimisation problem

- In the traditional *scenario*, generators control the network.
- Recently, utilisers can participate
  - by reducing (shaving) the load,
  - by shifting their utilisation in time, often combined to shaving,
  - and with similar manoeuvres.
- In the presence of distributed generation, a compound of utilisers can also play the role of a generator of relevant size: the term "prosumer" was introduced to indicate this double utiliser role (producer and consumer).
- The third class of actors comprehends national/regional Grid Managers or TSOs (Transmission System Operators) with the role of
  - maintaining continuous production/consumption balance (therefore controlling frequency),
  - ensuring the "rotating reserve" provision (some words on this later),
  - selecting the optimal generation distribution for each trading period,
  - consequently instructing generators about how much they have to produce, trading period per trading period.

The optimisation problem

- We know how a TSO can optimise a cost function network-wide (we did this for total cost via the KKT equations).
- We shall see later on (talking about load flow) that additional constraints cen be introduced to not overload transmission lines — but the *concept* does not change.
- We know how a generator/utiliser can determine its optimal operating conditions in a selfish manner.
- Determining the TSO decision is an extremely complicated procedure (and in fact, not strictly "automation").
- We provide a high-level overview, and then come back to control.

The optimisation problem

in a simplified world for clarity

- The TSO – or some other entity, either part of the TSO or coordinated, called in general the "electricity market" or "power exchange" or PX – comes to setting the energy price based on the bids to buy and the offers to sell (this is called short-term trading, we do not talk here about long-term operations).
- The TSO operates on a future trading period, knowing
  - a forecast of the demand (e.g., from historical data, meteo, and so forth),
  - the trading results for the period under question,
  - the generator characteristics (e.g., base load or available for slow or fast variations),
  - possible global requirements (cost, pollution, use of renewables, and so on).
- The TSO detremines for each generator
  - the production (constant) for the period, which is tertiary control,
  - and possibly the amount of additional production (rotating reserve) that the generator has to guarantee, via primary/secondary control, if needed to keep frequency at the set point; there is in general some remuneration for providing reserve, we omit details.

The resulting overall scheme

- Tertiary control results are sent to the generator controls as bias values, resulting in the scheme below (where we show that the secondary control distribution can be more complex than just a β_i, as we write for practical reasons in our exercises).

Electric systems – load flow

Foreword

- Prior to entering the subject, we need to recall/review two basic concepts:
    - how power is transferred in the network $\Rightarrow$ machine angle,
    - and how the effects of generators are combined $\Rightarrow$ network admittance matrix.
- Moreover, we need to abandon the purely energy-centred approach (no voltages or currents) taken so far, and adopt a phasor-oriented vision.
- Without impairing the conveyed message, we also recall/adopt the following simplifications:
    - single-voltage network (no transformers),
    - single-phase (or equivalently, perfectly balanced multiphase) system,
    - amplitude of generator voltages ideally controlled ,
    - *prevailing network*, i.e., all the generators are individually so small compared to the union of the others that each of them sees the network voltage as a fixed phasor.
- Of course the network frequency is controlled (now we know how).

Generator to network power transfer and machine angle

(review)

- Let $\underline{V}_n = V$ (phase 0) be the network voltage phasor.
- Let the generator voltage amplitude be V, thus its phasor be

$$\underline{V}_g = V\,(\cos\delta_{gn} + j\sin\delta_{gn})$$

  where δ_{gn} is the generator *machine angle* (w.r.t. the network).
- Let finally $\underline{Y}_{gn} = G_{gn} - jB_{gn}$ (mind the minus!) be the admittance of the generator–network connection.
- We already computed the active and reactive power (P_{gn} and Q_{gn}) flowing from generator to network, that is,

$$\begin{array}{rcl} P_{gn} & = & V^2\,(G_{gn}(1-\cos\delta_{gn}) + B_{gn}\sin\delta_{gn}) \\ Q_{gn} & = & V^2\,(B_{gn}(1-\cos\delta_{gn}) - G_{gn}\sin\delta_{gn}) \end{array}$$

Generator to network power transfer and machine angle

(review)

- Thus, varying δ_{gn} one can control P_{gn}, and Q_{gn} will follow as a consequence.
- To control *both* P_{gn} and Q_{gn} one may for example act on the excitation voltage. To see that, we can redo our computations with $|\underline{V}_g|$ set to V_{gm} instead of V.
- We already did this as well, and got

$$\begin{aligned} P_g &= G_{gn}V_{gm}^2 - V_{gm}V\left(G_{gn}\cos\delta_{gn} - B_{gn}\sin\delta_{gn}\right) \\ Q_g &= B_{gn}V_{gm}^2 - V_{gm}V\left(B_{gn}\cos\delta_{gn} + G_{gn}\sin\delta_{gn}\right) \end{aligned}$$

- In this course we do not deal with reactive power control.
- Suffice thus to say that to govern its power transfer to the network, a generator varies δ_{gn} with *transient* accelerations/decelerations...
- ...that however do not influence the network frequency, thus $\underline{V}_n$, owing to the prevailing network hypothesis.
- Actually there is a *local* influence at the swinging time scale, but this is not relevant for the problem we are treating now,

Node to node power transfer

and the network admittance matrix

- Obviously, if there is an electric connection between two nodes in a network, we can apply the same reasoning.
- The power transfer from node i to node j is computed exactly as we did for a generator to the network, coming to depend on $|\underline{V}_i|$, $|\underline{V}_j|$ and the angle δ_{ij} between the voltage phasors at the two nodes.
- Indicating the generic k-th node voltage magnitude $|\underline{V}_k|$ with V_k we thus have

$$
\begin{array}{rcl}
P_{ij} & = & g_{ij}V_i^2 - V_iV_j\left(g_{ij}\cos\delta_{ij} - b_{ij}\sin\delta_{ij}\right) \\
Q_{ij} & = & b_{ij}V_i^2 - V_iV_j\left(b_{ij}\cos\delta_{ij} + g_{ij}\sin\delta_{ij}\right)
\end{array}
$$

where $g_{ij} - jb_{ij}$ is the complex admittance of the line connecting nodes i and j (the reason for the lowercase g and b will be clarified soon).

Node to node power transfer

and the network admittance matrix

- Consider a network with n_B nodes or, to adopt the specific jargon of the addressed problems, n_B busses.
- Let $\underline{V}_i$ be the voltage at bus i and $\underline{I}_i$ the current injected in it by the locally connected generator(s) or drawn from it by the (local) bus load(s).
- In general each bus is connected to others via lines.
- Let $\underline{y}_{ij} = g_{ij} - jb_{ij}$ be the complex admittance of the line connecting busses i and j, of course with $i \neq j$.

Node to node power transfer

and the network admittance matrix

- Also, a bus may exhibit an admittance to ground.
- If this is true for bus i, we shall denote that admittance by $\underline{y}_{ii} = g_{ii} - jb_{ii}$.
- Some busses have at least one generator attached to them, and will be termed Generator (G) busses.
- The other busses carry only loads that absorb a certain amount of active power P and reactive power Q; these are called *Load* (L) busses or, more frequently, PQ busses.
- We need to describe the network by a matrix that will be useful for the problem we are discussing, and is called the admittance matrix.

The basics

Admittance matrix

- The admittance matrix $\underline{\mathbf{Y}}$ is created by
  - starting from the usual representation of a network containing *voltage* generators and *impedances* (there is a more synthetic formalism used in power network engineering, called the *one*- or *single-line diagram*, but we do not have the time to treat it),
  - injecting a *current* $\underline{I}_i$ in each bus i and computing the so induced voltages $\underline{V}_j$, in all the nodes,
  - indicating by $\underline{Y}_{ij}$ (uppercase, notice) the $\underline{I}_i/\underline{V}_j$ ratio,
  - and finally assembling $\mathbf{Y}$ as

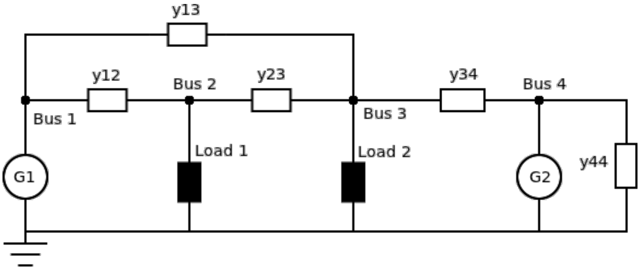
$$\underline{\mathbf{Y}} = [\underline{Y}_{ij}] = \begin{cases} \underline{y}_{ii} + \sum_{j=1, j\neq i}^{n_B} \underline{y}_{ij} & \text{for the diagonal elements,} \\ -\underline{y}_{ij} & \text{for the other elements,} \end{cases}$$

  - It should now be clear why we have used lowercase letters. Let us see an example to confirm our comprehension.

The admittance matrix

(an example)

- Consider the 4-busses network shown below:



- Remove all generators and loads, denote by $\underline{V}_i$ the voltage at the i-th bus, inject in each bus a current $\underline{I}_i$, write the nodal equations (KCL), and solve for the injected nodal currents.

The admittance matrix

(an example)

- Network with bus voltages and injected currents:

- Nodal equations (KCL) for busses 1–4:

$$
\begin{aligned}
\underline{I}_1 &= \underline{y}_{12}\left(\underline{V}_1-\underline{V}_2\right)+\underline{y}_{13}\left(\underline{V}_1-\underline{V}_3\right)\\
\underline{I}_2 &= \underline{y}_{12}\left(\underline{V}_2-\underline{V}_1\right)+\underline{y}_{23}\left(\underline{V}_2-\underline{V}_3\right)\\
\underline{I}_3 &= \underline{y}_{23}\left(\underline{V}_3-\underline{V}_2\right)+\underline{y}_{13}\left(\underline{V}_3-\underline{V}_1\right)+\underline{y}_{34}\left(\underline{V}_3-\underline{V}_4\right)\\
\underline{I}_4 &= \underline{y}_{34}\left(\underline{V}_4-\underline{V}_3\right)+\underline{y}_{44}\underline{V}_4
\end{aligned}
$$

- Note that obviously $\underline{y}_{ij}=\underline{y}_{ji}$, thus $\underline{\mathbf{Y}}$ is symmetric.

The admittance matrix

(an example)

- Now start from the KCLs

$$
\begin{aligned}
\underline{I}_1 &= \underline{y}_{12}\left(\underline{V}_1-\underline{V}_2\right)+\underline{y}_{13}\left(\underline{V}_1-\underline{V}_3\right)\\
\underline{I}_2 &= \underline{y}_{12}\left(\underline{V}_2-\underline{V}_1\right)+\underline{y}_{23}\left(\underline{V}_2-\underline{V}_3\right)\\
\underline{I}_3 &= \underline{y}_{23}\left(\underline{V}_3-\underline{V}_2\right)+\underline{y}_{13}\left(\underline{V}_3-\underline{V}_1\right)+\underline{y}_{34}\left(\underline{V}_3-\underline{V}_4\right)\\
\underline{I}_4 &= \underline{y}_{34}\left(\underline{V}_4-\underline{V}_3\right)+\underline{y}_{44}\underline{V}_4
\end{aligned}
$$

- and express $\underline{\mathbf{Y}}$, obtaining

$$
\begin{bmatrix}\underline{I}_1\\ \underline{I}_2\\ \underline{I}_3\\ \underline{I}_4\end{bmatrix}
=
\begin{bmatrix}
\underline{y}_{12}+\underline{y}_{13} & -\underline{y}_{12} & -\underline{y}_{13} & 0\\
-\underline{y}_{12} & \underline{y}_{12}+\underline{y}_{23} & -\underline{y}_{23} & 0\\
-\underline{y}_{13} & -\underline{y}_{23} & \underline{y}_{13}+\underline{y}_{23}+\underline{y}_{34} & -\underline{y}_{34}\\
0 & 0 & -\underline{y}_{34} & \underline{y}_{34}+\underline{y}_{44}
\end{bmatrix}
\begin{bmatrix}\underline{V}_1\\ \underline{V}_2\\ \underline{V}_3\\ \underline{V}_4\end{bmatrix}
$$

- Verify the rule and convince yourselves:

$$
\underline{\mathbf{Y}}=\left[\underline{Y}_{ij}\right]=\begin{cases}
\underline{y}_{ii}+\sum_{j=1,j\neq i}^{n_B}\underline{y}_{ij} & \text{for the diagonal elements,}\\
-\underline{y}_{ij} & \text{for the other elements,}
\end{cases}
$$

The admittance matrix

- The admittance matrix can also be used to compute the power injected in all busses if voltages are known, as apparently

$$\underline{V} = \underline{\mathbf{Y}}^{-1}\underline{I}$$

where $\underline{V}$ and $\underline{I}$ are respectively the vectors of bus voltages and *injected* currents; matrix $\underline{\mathbf{Y}}^{-1}$ is also called the network (or nodal, or bus) *impedance* matrix, and denoted by $\underline{\mathbf{Z}}$.

- Therefore, knowing the bus voltage phasors, one can obtain the complex power injected at each bus as

$$\underline{S} = \underline{V} \circ \underline{I}^* = \underline{V} \circ (\underline{\mathbf{Y}}\,\underline{V})^*$$

where $\circ$ denotes the Schur product, and * the complex conjugate.

- We shall soon go through another example, to help us formulate the Load Flow (LF) problem.

Network-related problems

Load Flow and Optimal Power Flow

- So far, we have been dealing with optimising the generation of the required power.
- However, generating here or there entails using the transmission lines differently.
- We need to ensure that no line gets overloaded, and if possible to account – when computing the generation cost – also for the power lost in the transmission process itself.

- In other words, we need to check how power "flows" in the network.
- This problem has been historically given two names:
  - the Load Flow (LF) problem, on which we are saying some words,
  - and the Optimal Power Flow (OPF) problem, that in this course we shall just mention.

Load Flow

An introductory example

- Consider a network with two busses 1 and 2.
- Take $\underline{V}_2$ as phase reference (i.e., assume its phase is zero).
- Let busses 1 and 2 be connected by a line of complex admittance $\underline{y}_{12} = g_{12} - jb_{12}$.
- Denote by δ_{12} the difference between the phases of $\underline{V}_1$ and $\underline{V}_2$ – i.e., the phase of $\underline{V}_1$ is δ_{12} – and go for Maxima:

```
y12 : g12-%i*b12;
Y   : matrix([y12,-y12],[-y12,y12]);
V   : matrix([V1*(cos(d12)+%i*sin(d12))],[V2]);
I   : Y.V;
S   : V*conjugate(I);
P   : trigsimp(realpart(S));
Q   : trigsimp(imagpart(S));
```

Load Flow

An introductory example

- We obtain

$$
\begin{aligned}
P_1 &= g_{12}V_1^2 - V_1V_2\left(g_{12}\cos\delta_{12} - b_{12}\sin\delta_{12}\right) \\
Q_1 &= b_{12}V_1^2 - V_1V_2\left(b_{12}\cos\delta_{12} + g_{12}\sin\delta_{21}\right) \\
P_2 &= g_{12}V_2^2 - V_1V_2\left(g_{12}\cos\delta_{12} + b_{12}\sin\delta_{12}\right) \\
Q_2 &= b_{12}V_2^2 - V_1V_2\left(b_{12}\cos\delta_{12} - g_{12}\sin\delta_{12}\right)
\end{aligned}
$$

- These four equations provide $P_{1,2}$ and $Q_{1,2}$, i.e., the (signed) active and reactive power injected at each of the two busses, based on the knowledge of their voltage phasors (and of course of the network parameters).

- Now, let us view the problem in another way.

Load Flow

The problem *per exemplum*

- Suppose that bus 2 is a Load (PQ) bus, and bus 1 a Generator (G) one.
- Suppose to know the active power demand from the load (remember the forecasts for tertiary control?) and that loads are managed so that the reactive demand is maintained within a prescribed power factor.
- Suppose, in one word, to know P_2 and Q_2.
- Suppose then that the *active* power generation at bus 1 is controlled (remember how P_m was managed to match P_e via power/frequency control?) and the same is true for the voltage *magnitude* V_2—not for the phase, as this is the means to release power.
- Again, take the phase of $\underline{V}_2$ as reference, and – remember – assume ideal (or almost ideal) frequency control.

Load Flow

The problem *per exemplum*

- Given all the above, we are dealing again with the equations

$$\begin{array}{rcl} P_1 & = & g_{12}V_1^2 - V_1V_2\left(g_{12}\cos\delta_{12} - b_{12}\sin\delta_{12}\right) \\ Q_1 & = & b_{12}V_1^2 - V_1V_2\left(b_{12}\cos\delta_{12} + g_{12}\sin\delta_{21}\right) \\ P_2 & = & g_{12}V_2^2 - V_1V_2\left(g_{12}\cos\delta_{12} + b_{12}\sin\delta_{12}\right) \\ Q_2 & = & b_{12}V_2^2 - V_1V_2\left(b_{12}\cos\delta_{12} - g_{12}\sin\delta_{12}\right) \end{array}$$

  but in the three unknowns Q_2, δ_{21}, V_2.
- Note that also in the previous view the free quantities were in fact three, as V_1, V_2 and δ_{12} decided all of the rest.
- To determine all the nodal voltage phasors, thus, we could write
  - the balance equation for active and reactive power at bus 1 (PQ),
  - and the balance equation for active power at bus 2 (G).
- This is a nutshell-size example of LF problem. Let us now abstract and generalise.

Load Flow

Problem statement

- Given a network composed of
  - n_G generator (G) busses,
    where the (injected) active power and the voltage amplitude are known,
  - plus n_{PQ} load (PQ) busses,
    where the (drawn) active and reactive powers are known,
  - plus one bus, called the *slack* (S) bus,
    where the voltage amplitude and phase are known,

  determine all the voltage phasors (magnitudes and phases).

- Note: of course $n_G + n_{PQ} + 1$ equals the total number n_B of busses, but in the following we shall preferably count busses by type.

Load Flow

Unknowns, equations, and solution

- Unknowns:
  - $2(n_G + n_{PQ} + 1)$, i.e., all voltage phasors' magnitudes and phases
  - minus n_G because at G busses the voltage amplitude is known
  - minus 2 because at the S bus both voltage amplitude and phase are known,

  $\Rightarrow$ for a total of $n_G + 2n_{PQ}$.
- Equations:
  - n_G balances of active power at G busses,
  - plus n_{PQ}, balances of active power at PQ busses
  - plus n_{PQ}, balances of reactive power at PQ busses,

  $\Rightarrow$ for a total of $n_G + 2n_{PQ}$.
- Solution:
  - the problem is not dynamic, apparently,
  - but at the same time highly nonlinear;
  - many numerical methods were proposed and are continuously studied,
  - ranging from standard ones (e.g., Newton-Raphson) or modifications of these, up to completely *ad hoc* ones (not the matter of this course).

Load Flow

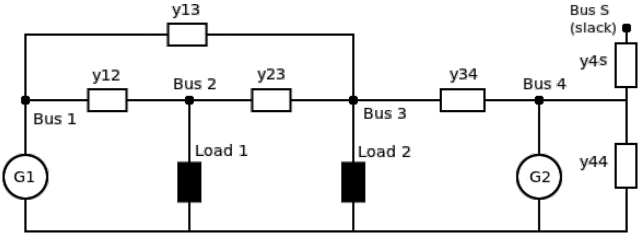
The slack bus

- The slack bus, can be just considered a reference, *mutatis mutandis* pretty much like the ground when solving an electric circuit,
- Alternatively, it can be viewed as the connection to a larger – e.g., cross-national – network, that is seen as a fixed phasor since such a network can be assumed to have prevailing power w.r.t. the considered – e.g., national – one, like the considered one is assumed to have w.r.t. any individual generator in it.

Load Flow

An example

- Let us write the LF equations for the network



Load Flow

An example – solution

- First (although not strictly necessary, one could directly reason with its elements) write the admittance matrix:

$$
\underline{\mathbf{Y}} = \begin{bmatrix}
\underline{Y}_{11} & \underline{Y}_{12} & \underline{Y}_{13} & \underline{Y}_{14} & \underline{Y}_{1s} \\
\underline{Y}_{12} & \underline{Y}_{22} & \underline{Y}_{23} & \underline{Y}_{24} & \underline{Y}_{2s} \\
\underline{Y}_{13} & \underline{Y}_{23} & \underline{Y}_{33} & \underline{Y}_{34} & \underline{Y}_{3s} \\
\underline{Y}_{14} & \underline{Y}_{24} & \underline{Y}_{34} & \underline{Y}_{44} & \underline{Y}_{4s} \\
\underline{Y}_{1s} & \underline{Y}_{2s} & \underline{Y}_{3s} & \underline{Y}_{4s} & \underline{Y}_{ss}
\end{bmatrix}
$$

$$
= \begin{bmatrix}
\underline{y}_{12}+\underline{y}_{13} & -\underline{y}_{12} & -\underline{y}_{13} & 0 & 0 \\
-\underline{y}_{12} & \underline{y}_{12}+\underline{y}_{23} & -\underline{y}_{23} & 0 & 0 \\
-\underline{y}_{13} & -\underline{y}_{23} & \underline{y}_{13}+\underline{y}_{23}+\underline{y}_{34} & -\underline{y}_{34} & 0 \\
0 & 0 & -\underline{y}_{34} & \underline{y}_{34}+\underline{y}_{44} & -\underline{y}_{4s} \\
0 & 0 & 0 & -\underline{y}_{4s} & \underline{y}_{4s}
\end{bmatrix}
$$

Load Flow

An example – solution (Maxima, we are not doing this by hand)

```
y12   : g12-%i*b12;
y13   : g13-%i*b13;
y23   : g23-%i*b23;
y34   : g34-%i*b34;
y44   : g44-%i*b44;
y4s   : g4s-%i*b4s;
Y     : matrix([ y12+y13, -y12    , -y13         ,  0      ,  0  ],
               [-y2     ,  y12+y23, -723         ,  0      ,  0  ],
               [-y13    , -y23    ,  y13+y23+y34, -y34    ,  0  ],
               [ 0      ,  0      , -y34         ,  y34+y44, -y4s],
               [ 0      ,  0      ,  0           , -y4s    ,  y4s]);
V1    : V1m*(cos(d1)+%i*sin(d1)); /* unknown  d1, V1m is known (G bus) */
V2    : V2m*(cos(d2)+%i*sin(d2)); /* unknowns V2m and d2 (PQ bus)      */
V3    : V3m*(cos(d3)+%i*sin(d3)); /* unknowns V3m and d3 (PQ bus)      */
V4    : V4m*(cos(d4)+%i*sin(d4)); /* unknown  d4, V4m is known (G bus) */
Vs    : Vsm;                      /* phase 0, slack is reference       */
V     : transpose(matrix([V1,V2,V3,V4,Vs]));
I     : Y.V;                 /* matrix product (dot) here              */
S     : V*conjugate(I);      /* element by element product (star) here */
P     : trigreduce(realpart(S));
Q     : trigreduce(imagpart(S));
LFeq1 : P1 = P[1,1];
LFeq2 : P4 = P[4,1];
LFeq3 : P2 = P[2,1];
LFeq4 : Q2 = Q[2,1];
LFeq5 : P3 = P[3,1];
LFeq6 : Q3 = Q[3,1];
```

...but given a network, you are expected to know *which* are the LF equations.

Load Flow

Role in the overall network control

- Once primary/secondary control is in place and tertiary optimisation is done, use LF to check that no line is overloaded by also computing currents. If said check fails, modify the optimised solution to a suboptimal one "near" to the optimal but fulfilling the overload avoidance constraints.
- Express the mentioned overload avoidance conditions and plug them into tertiary optimisation as additional constraint. Note that this *significantly* complicates the optimisation problem on the constraint side.
- Use LF to compute a transmission cost in terms of power lost over the lines, and plug this into tertiary optimisation. Doing so further complicates the optimisation, this time also by modifying the cost function.

A general *panorama*

on control in electric (AC) systems

- We now have a reasonably complete *panorama* of network control and optimisation for the purpose of this course.
- Our boundaries are generator internals "below" and management optimisation "above": automation and control, that we discussed, lies in the middle:
  - primary control – proportional to frequency error, act in seconds or tens of, local, stop frequency drift;
  - secondary control – integral, steer frequency back to nominal, act in minutes, computed at network level and distributed;
  - tertiary control: optimise generation distribution and transmission, account for economic facts (when it is to control frequency, physics rules), computed and actuated by trading periods (in the order of 1/2 hour).

Part C

Thermal systems

Thermal systems – generalities

System characteristics

Main problems

Foreword

- As we saw (in E years) or shall see later on (in T years), electric (AC) systems
  - are characterised by network-wide interaction, as frequency is an inherently network-wide quantity;
  - exhibit very fast coupling (wrt control bands) among the network nodes;
  - allow for a modest role of storage, at least at the network level;
  - are governed with highly standardised control schemes, that involve a few strong central authorities.
- Thermal systems are radically different:
  - interaction is much looser, and mostly localised to well identifiable parts of the system;
  - tendentiously slow coupling (again, wrt control bands) is observed;
  - the role of storage can be paramount in several system components;
  - control schemes are very heterogeneous, and often contain numerous local authorities with limited power (*lato sensu*).

Foreword

- As a first consequence, wrt the electric case,
  - there are more concurrent time scales to consider,
  - which helps establish some hierarchy in the systems to address
  - but same time results in a higher structural complexity.
- As a second consequence, and again wrt electric systems,
  - we are encountering a wider variety of problems
  - and thus we need a larger zoo of models.

- This motivates the co-existence of the two system types (E and T) in this course...
- ...though we can devote just a limited amount of time to talk about their coordination.

Thermal systems – component models

Introduction and hypotheses

Fluid transport and flow control components

Heat exchangers

Thermal machines

Containment elements

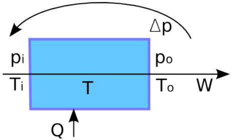
Main elements to be modelled

- Fluid (liquid) transport, motion and flow control (pipes, pumps, valves);
- heat exchangers (liquid/liquid and liquid/air);
- thermal machines (boilers, chillers, heat pumps);
- containment elements and associated thermal exchanges (walls, openings, conductive/convective/radiative exchanges).
- Note: for simplicity, unidirectional fluid flow (quite realistic, anyway) will be assumed.

Involved substances and materials

- We shall treat thermo-vector fluids (e.g., water flowing in heating elements) as incompressible, and with constant specific heat;
- the same will be done for the air contained in buildings (pressure is practically the atmospheric one):
- thus, we shall not treat centralised air treatment systems like AHUs (Air Handling Units) and air-based heat distribution — if not very marginally.
- Solid materials will come into play for containments, and here too simple descriptions (constant and uniform properties) will be used:
- always recall our system-level attitude.

Pipes



- Hydraulic equation (review, Δz is the height difference (inlet minus outlet), ρ the fluid density):

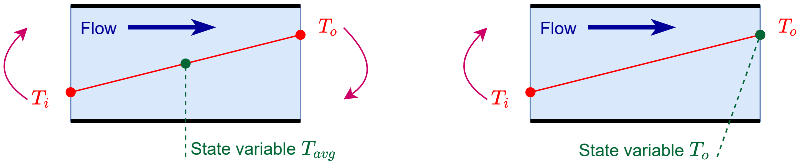
$$\Delta p = \frac{K_T}{\rho} w^2 - \rho g \Delta z$$

- Energy balance (T assumed uniform and equal to T_o, V is the pipe volume, c the fluid specific heat):

$$\rho V c \dot{T}_o = c w (T_i - T_o) + Q$$

Pipes

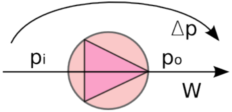
Why T_o as state variable — connected to our unidirectional flow hypothesis

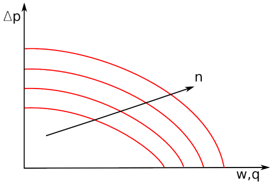


- Which T shall we take to represent the lump energy content?
- Suppose we take the average T_{avg} (left) and then apply a step-like variation to the inlet one T_i:
  - *natura non facit saltus — ergo nec faciunt status variabiles* ☺ (nature does not make jumps — hence neither state variables do)

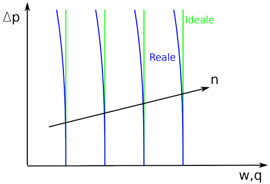
  $\Rightarrow$ we would see a transient negative variation of the outlet temperature T_o, i.e., a non-physical nonminimum-phase behaviour.
- Apparently this does not happen (right) if we take T_o as state variable (but we have to know/assume the flow direction).

Pumps





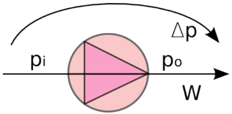
Centrifugal



Volumetric

- The command n prescribes the pump speed, typically in rpm.
- Mass and volume flowrate (here proportional by ρ) are denoted by w and q, respectively.

Pumps



- Centrifugal pump hydraulics (given the fluid):

$$\Delta p = H_0(n) - H_1(n) w^2$$

where, given a nominal rpm n_0 and correspondingly $H_0 = \overline{H}_0$ e $H_1 = \overline{H}_1$,

$$H_0(n) = \overline{H}_0 \, n/n_0, \quad H_1(n) = \overline{H}_1 \, n/n_0.$$

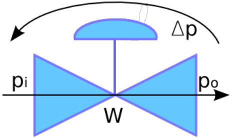
- Volumetric pump hydraulics (given the fluid):

$$w = Kn$$

where K is a characteristic parameter.

- We can neglect mechanical heating, thus in both cases $T_o = T_i$.

Valves



- Hydraulic equation (we assume proper sizing, the curious can refer e.g. to ANSI/ISA-75.01.01 or IEC 60534-2-1):

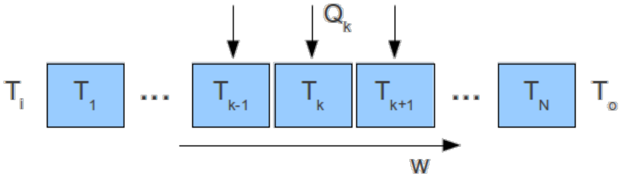
$$w = Cv_{max}\Phi(x)\sqrt{\rho\Delta p}$$

- Isenthalpic transformation:

$$h_o = h_i.$$

Fluid streams

Lumped (Finite Volumes) model



- For incompressible fluids, thermal equations are decoupled from hydraulics (constant density).
- Having decided the flow direction, the equation for each element of the FV description is thus

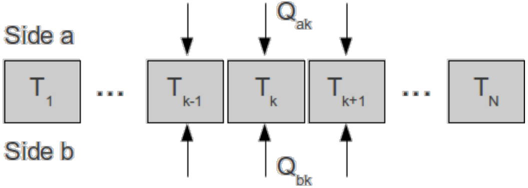
$$\rho c V_k \dot{T}_k = wcT_{k.1} - wcT_k + Q_k, \qquad T_{-1} = T_i,\ T_o = T_N,$$

where the V_k are the lump volumes, often (for us, always) equal.

Main geometries

- Plate, shell-tube...
- All composed of two fluid streams separated by a wall, typically made of well-conducting metal.

Wall

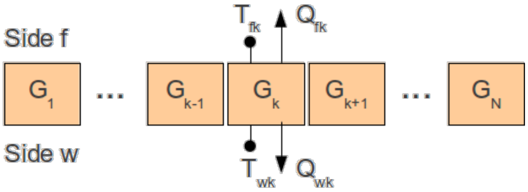


- Dividing the wall in the same way as the stream(s),

$$\rho_w c_w V_{wk} \dot{T}_k = Q_{ak} + Q_{bk}$$

  where the "w" subscript stands for "wall".
- Axial conduction in the wall is neglected, which is realistic enough for our purposes.

Convective exchanges



- Finally comes the (convective) exchange element, that adopting the same discretisation as before yields

$$Q_{fk} = -Q_{wk} = \gamma S_k (T_{fk} - T_{wk})$$

  where the f and w subscript stand for "fluid" and "wall".
- The S_k are the lump surfaces while γ is the thermal exchange coefficient, either constant or dependent on the flow conditions.

Convective exchanges

Correlations for γ

- In the literature many correlations to conpute γ were proposed.
- These typically distinguish
  - natural convection, caused by buoyancy, vs. forced convection, induced by movers like pumps and fans;
  - laminar flow, with particles moving "by parallel lanes" and prabolic velocity profile on the duct cross surface, vs. turbulent flow, with particles mixing randomly and a quasi-flat velocity profile;
  - intuitively, forced turbulent convection is most efficient.
- In the forced case the kind of flow is discriminated by the Reynolds number

$$Re = \frac{\rho u D}{\mu}$$

  where ρ is density, u velocity, D (hydraulic) diameter, and μ dynamic viscosity.
- Low Re means laminar flow, high Re turbulent flow (there is debate on thresholds, will show some data in exercises).

Convective exchanges

Correlations for laminar and turbulent flow

- A notorious example is the Dittus-Bölter correlation for turbulent flow

$$Nu = 0.023\,Re^{4/5}\,Pr^{n}$$

where Nu and Pr are respectively the Nusselt and Prandtl numbers

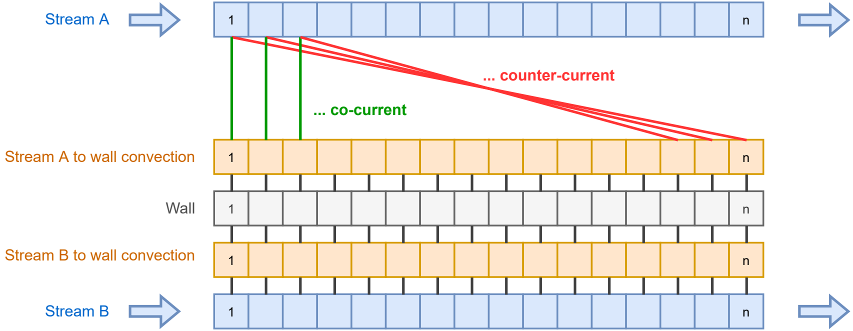
$$Nu = \frac{\gamma L}{\lambda}, \quad Pr = \frac{c_p \mu}{\lambda}$$

with L characteristic dimension (in tubes, stream diameter), λ thermal conductivity, and c_p specific heat; n is 0.3 for the fluid being heated, 0.4 if the fluid is cooled.

- Further details would stray from our scope; the curious can check out e.g. `https://en.wikipedia.org/wiki/Nusselt_number` for the variety of cases.

Complete models

typical liquid-liquid case



- The connections realise the heat exchanger configuration (e.g., as shown, co-current or counter-current).
- Complex geometries may require more articulated interconnections (not our subject).

Complete models

One-volume liquid-liquid case

- In large models (e.g., a complete heat network) one may sometimes not be interested in the detailed dynamics of every (substation) heat exchanger.
- On the other hand, in addition, one may want to parametrise exchanger models with minimal – most typically, datasheet – information, such as
  - nominal hot and cold side flowrates,
  - nominal hot and cold side inlet temperatures,
  - nominal transferred power and efficiency.
- In such cases, always keeping an eye on validity limits as already noted, very simple models can be devised.
- We show a quite extreme case, with one volume (per side) only.

Complete models

One-volume liquid-liquid case (same fluid on both sides)

- Maxima:

```
/* Given c,wh,wc,Thi,Tci,P,eta    (sp. heat, flowrates, inlet Ts, nom. P, efficiency)  */
   e1: c*wh*(Thi-Tho) = P;        /* heat taken from (h)ot side                        */
   e2: c*wc*(Tco-Tci) = P*eta;    /* heat released to (c)old side (1-eta is p.u. loss) */
   e3: P=G*(Thi-Tci);             /* G is equivalent conductance -- question: why "i"? */
   solve([e1,e2,e3],[G,Tho,Tco]);
```

- whence, introducing hot and cold side volumes V_h, V_c to compute heat capacities,

$$
\begin{aligned}
P_h &= G(T_{hi}-T_{ci})\\
P_c &= \eta P_h\\
\rho c V_h \dot{T}_{ho} &= c w_h (T_{hi}-T_{ho}) - P_h\\
\rho c V_c \dot{T}_{co} &= c w_c (T_{ci}-T_{co}) + P_c
\end{aligned}
$$

Complete models

Another extreme case (just some words)

- In some cases it can be convenient to have only one volume on one side, while the other is still represented by a sequence of volumes along its stream.
- A couple of examples:
  - when one side is a.g. a serpentine and the other a storage tank that can be taken as fully mixed;
  - when one side has a very small fluid residence time (contained mass over traversing flowrate) like e.g. the air volume in a fan coil.
- In such cases the model still has the structure shown previously for the two-stream exchanger, however with the convective conductances on the single-volume stream side all connected to that volume.

A broad classification and two hypotheses

- For our purposes, we distinguish two types of machines:
  1. those that inject heat into a fluid by combustion or from analogous sources, such as solar radiation, that can possibly be modulated (e.g., by operating a fuel valve or focusing/defocusing a mirror); these include boilers, thermal solar captors, and similar objects;
  2. those that employ work to transfer heat from a cold to a hot source; these include all kinds of heat pumps.
- Hypotheses:
  1. at a system level, thermal machines can be described as static relationships coupled to a simple (for us, 1st-order) dynamics to represent the internal mass/energy storage, or in other words connected to the fluid residence time (contained mass over traversing flowrate, recall);
  2. at the same level, the fluid(s) perceive the machine operation as impressed heat rates: possible relationships of those heat rates with temperatures can be represented as static characteristics.

Thermal machines

Type 1 machines – example: heater, "direct" model

- Denoting by w the fluid flowrate, by w_f the fuel one, by HH its calorific power, by V the volume of contained fluid and by η_c a "combustion" efficiency, one can simply write with self-explanatory notation

$$\rho c V \dot{T}_o = wc(T_i - T_o) + w_f HH \eta_c(w, T_o)$$

  where $\eta_c(w, T_o)$, or just $\eta_c(w)$ depending on the detail level, provides the machine efficiency curve.
- Suggestion: try to reformulate for a thermal solar captor where the captured radiative flux can be partialised.

Thermal machines

Type 1 machines – example: heater, "inverse control-based" model

- Sometimes models like the above are too simplistic, yet physically detailed ones are too demanding and/or the information required for them is not available.
- In such cases, if the object to model has controls aboard. one can exploit them...

  ... in the way that we show her for a heater. but is clearly more general.

- Idea: trust the control designer and assume that the controlled variable y tracks the set point w through a reasonably low-order dynamics; the ideal case is

$$\frac{Y(s)}{W(s)} = \frac{1}{1+s/\omega_c}$$

  where to know ω_c it suffices to know the closed-loop settling time (from technical specs or even just logged signals).

- Now let us apply the idea to the water heater.

Thermal machines

Type 1 machines – example: heater, "inverse control-based" model

- Heater on $\Rightarrow$ outlet temperature T_o follows set point T_{set} with closed-loop time constant τ_{cl}, thus
$$T_o + \tau_{cl}\dot{T}_o = T_{set}.$$
- Heater off $\Rightarrow$ outlet temperature T_o approaches inlet one T_i with free-cooling time constant τ_{fc}, thus
$$T_o + \tau_{fc}\dot{T}_o = T_i.$$
- Summing up, since both T_i and the flowrate w will be dictated by the connected thermo-hydraulic network, the model just requires τ_{cl} and τ_{fc}, and reads

$$\begin{aligned}
\dot{T}_o &= \begin{cases} -\frac{1}{\tau_{cl}}T_o + \frac{1}{\tau_{cl}}T_{set} & \text{heater on} \\ -\frac{1}{\tau_{fc}}T_o + \frac{1}{\tau_{fc}}T_i & \text{heater off} \end{cases} \\
P_h &= \begin{cases} wc(T_o - T_i) & \text{heater on} \\ 0 & \text{heater off} \end{cases}
\end{aligned}$$

Thermal machines

Type 1 machines – example: heater, "inverse control-based" model

- Evident limits:
  - knowing τ_{cl} provides information about the loop transfer function, not on C and P, so that we may represent well the controlled temperature but not the control signal, hence power P_h;
  - in addition, the idea just introduced and exploited lives entirely in a linear context, so that we have no means to account for saturation limits on P_h to the further detriment of its representation.
- We could however do better with some knowledge of the controlled process: let us therefore assume to know the maximum heater power $P_{h,max}$, the contained fluid volume V, and a nominal/design fluid flowrate w — not too tall an order at all with heater datasheet and installation conditions at hand.

Thermal machines

Type 1 machines – example: heater, "inverse control-based" model

- Energy balance in the "heater on" case:

$$c\rho V\dot{T}_o = wc(T_i - T_o) + P_h.$$

- This is nonlinear (w times T), but let us assume w controlled (i.e., input) and seldom changed, so as to take it as a parameter and write for the process

$$T_o(s) = \frac{1}{1+s\,{}^{\rho V}\!/_{w}} T_i(s) + \frac{{}^{1}\!/_{wc}}{1+s\,{}^{\rho V}\!/_{w}} P_h(s).$$

- Note that increasing w decreases both the $P_h \to T_o$ gain ${}^{1}\!/_{wc}$ and the process time constant ${}^{\rho V}\!/_{w}$ — not surprisingly, the residence time.

Thermal machines

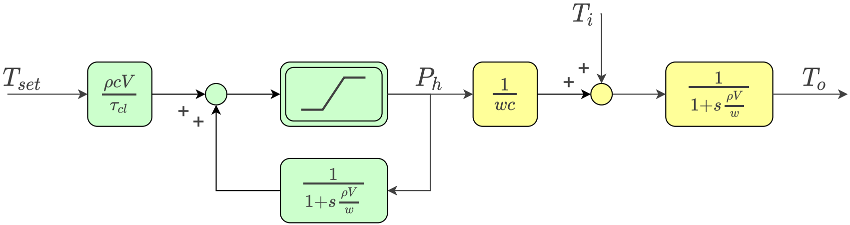
Type 1 machines – example: heater, "inverse control-based" model

- Up to here, almost objective.
- Now comes the controller, and here we have to make some further assumptions:
  - we take a PI – very reasonable – and assume the integral time equal to the process time constant (it might be a little smaller, we are not discussing this now),
  - while the gain is easy to guess as we know the closed-loop time constant;
  - most important, then, we have to choose an antiwindup mechanism;
  - this is in fact really guessing and we are not deepening the issue here, if not for noticing that in the absence of datasheet information recorded data can help;
  - to exemplify, in the following we use the internal feedback technique.

Thermal machines

Type 1 machines – example: heater, “inverse control-based” model

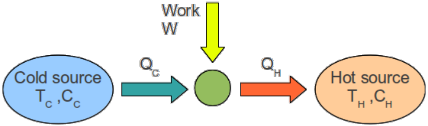
- Resulting block diagram in the “heater on” case:



- The saturation block has limits 0 and $P_{h,max}$.
- The model – including the “heater off” case – is easy to write in the form of conditional equations: in the “off” case it suffices to set the “fake PI” (green blocks) to tracking at zero output, while the process state is always T_o.

Thermal machines

Type 2 machines – example: heat pump



- The steady-state balance is apparently $Q_H = Q_C + W$, where W can come e.g. from a compressor (we do not treat more articulated cases such as absorption cycles).
- Machines like the one here schematised are statically described by the so called *Coefficient Of Performance* (COP), defined as"useful effect over needed work". As one may use the machine for heating or cooling, we have

$$COP_{heat} = \frac{Q_H}{W}, \quad COP_{cool} = \frac{Q_C}{W}$$

where

$$\frac{Q_H}{W} = \frac{Q_C + W}{W} \quad \Rightarrow \quad COP_{heat} = COP_{cool} + 1.$$

Thermal machines

Type 2 machines – example: heat pump

- If the machine operates at the maximum theoretical (Carnot) efficiency

$$\frac{Q_H}{T_H} = \frac{Q_C}{T_C}.$$

- Thus, theoretical (Carnot) values for COP_{heat} and COP_{cool} can be defined as

$$COP^C_{heat} = \frac{T_H}{T_H - T_C}, \quad COP^C_{cool} = \frac{T_C}{T_H - T_C},$$

  where, remember, absolute (Kelvin) temperatures are to be used.

- To represent real machines efficiencies are introduced, hence

$$COP_{heat} = \eta_h \frac{T_H}{T_H - T_C}, \quad COP_{cool} = \eta_c \frac{T_C}{T_H - T_C},$$

  where η_h and η_c, $0 < \eta_{h,c} < 1$, can be considered constant (as we shall do) or be made dependent on temperatures.

Thermal machines

Type 2 machines – example: heat pump

- To obtain a simple (system-level) dynamic model, associate two thermal capacities $C_{H,C}$ to the hot and cold sources (i.e., considering the classical refrigerator cycle as a representative example, to the condenser and the evaporator respectively).
- This, together with the previous COP considerations, yields for the heating case

$$
\begin{aligned}
C_H \dot{T}_H &= Q_H + Q_{Heh} \\
C_C \dot{T}_C &= Q_C + Q_{Cec} \\
COP_{heat} &= \eta_h \frac{T_H}{T_H - T_C} \\
Q_H &= COP_{heat} W \\
Q_H &= Q_C + W
\end{aligned}
$$

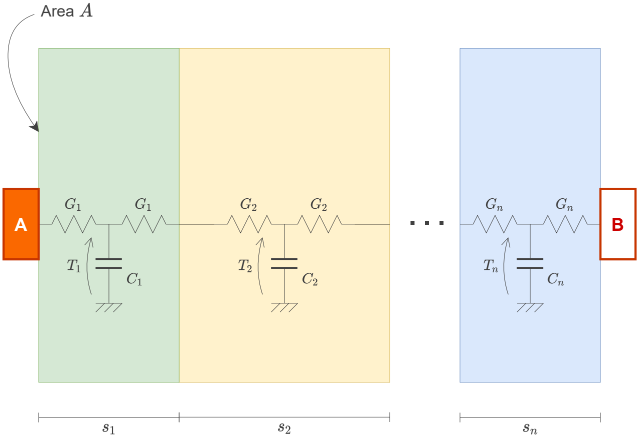
where Q_{Heh} and Q_{Cec} are the heat rates exchanged by the H and C sides (typically by convection) with the environments to which either of the two is exposed; W is here an input.

Preliminaries

- Containment elements are simple conduction ones.
- Such elements are possibly (and in fact most frequently) composed of a series of layers of different materials (i.e., with different specific heats and conductivities)...
- ...so that their electric equivalent is a series of RC cells.
- Glazing in buildings entails many particular cases and may require specific modelling (for example, windows often need to account for the thermal power transmitted by radiation), but we do not deal with such details in this course.

A multilayer wall

— nonhomogeneous case, planar geometry only for simplicity



$$C_i = c\rho A s_i, \quad G_i = \lambda \frac{A}{s_i/2}$$

An interesting particular case

- A specific case of through-containment heat exchange is air renovation, on which we conversely spend some words.
- Denoting by w_r the renovation air flow rate, and with T_i and T_e the internal and external temperatures, respectively, we have

$$\begin{aligned} Q_{e\to i} &= w_r c_a T_e \\ Q_{i\to e} &= w_r c_a T_i \end{aligned}$$

  where c_a is the air specific heat.
- Summing the above with the due signs, the net e→i heat rate is

$$Q_{ei} = G_{ei}(T_e - T_i), \quad G_{ei} = w_r c_a$$

  where G_{ei} plays the role of an equivalent thermal conductance, governed by the renovation flowrate.

Thermal systems – control problems (part 1)

Flow/pressure control with liquids

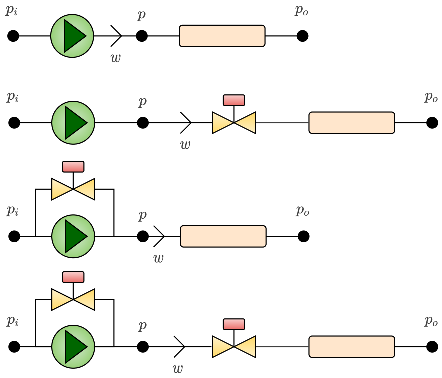
Temperature control in pipes

Joint temperature and flow control

Foreword

- In thermal energy systems flow control is related to “moving energy around” properly, by transporting fluids.
- Pressure control is mostly introduced as a means to not exceed equipment limits.
- In this course we restrict the scope to liquid thermovector fluids.
- The involved process components are piping, pumps and valves (the latter two as actuators).
- Our main point is that these introduce nonlinear characteristics in control loops.
- We need to manage this (and say a few words on sizing).

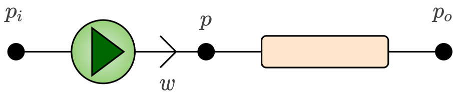
Problem statement



- The main schemes build upon
  - variable-speed pump,
  - fixed-speed pump with series valve,
  - fixed-speed pump with recirculating valve.
- The main sources of disturbance are
  - variations in the inlet and/or the outlet pressure (for the shown open circuit case),
  - variations in the load hydraulic impedance.

Problem statement

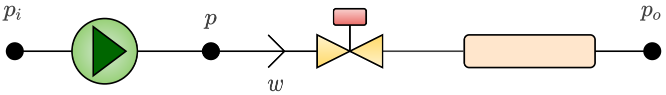
Typical configuration a



- Need variable-speed pump:
  - can control either p or w;
  - relief lines are provided if p could – but must not – exceed some limits.

Problem statement

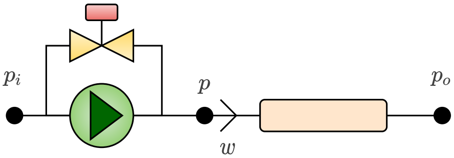
Typical configuration b



- Fixed-speed pump (most commonly used):
  - valve opening can control either p or w;
  - sometimes override control is used: keep w if p does not exceed a maximum, else keep p.
- Variable-speed pump:
  - use pump speed to control p and valve opening to control w;
  - possibly adjust the p set point to keep the valve opening in range (say 20–80%) while keeping pump speed – thus power – as low as possible.

Problem statement

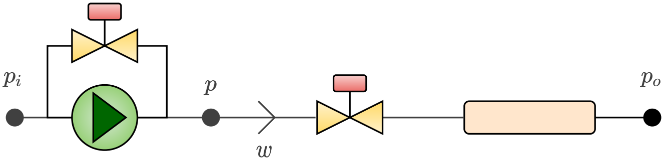
Typical configuration c



- Fixed-speed pump (most frequent case):
  - valve opening typically used to control w;
  - relief lines are provided for p if needed.
- Variable-speed pump (quite uncommon):
  - use pump speed and valve opening to control both p and w;
  - coupling between the two is a bit complex, however.

Problem statement

Typical configuration d



- Fixed-speed pump (hardly any use for a variable-speed one):
  - recirculating valve opening typically used to control p,
  - while line valve opening controls w.

Problem statement

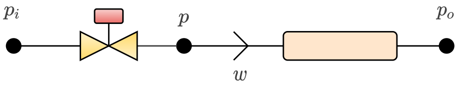
Common facts and wrap-up

- All the relevant dynamics is in the actuator (pump motor and/or valve positioner);
- the said dynamics is reasonably linear at least in the small (rate limits in the large);
- hence the main issue is the nonlinearity of pump and/or valves, as well as of the hydraulic load,
- which collectively result in the controller(s) seeing a variable differential gain cascaded to a quite low-order linear dynamics (1st–2nd normally suffices).

- For space/time reasons we are thus studying here just a couple of representative cases: generalisation to the variety of possible ones is up to the engineer.
- Then we are solving a few control problems for completeness.

Case 1

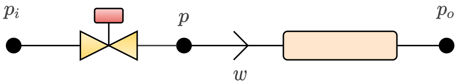
Setting



- Assignment:
  1. size the valve flow coefficient;
  2. determine the valve installed characteristic;
  3. make the said characteristic linear (physically or in software).
- We exploit this first case to also provide the necessary definitions "on the job".

Case 1

Setting



- Hypotheses:
  - p_i and p_o are prescribed boundary conditions;
  - the valve (for liquid, v subscript) is ruled by

$$w_v = C_v \phi(x) \sqrt{\rho \Delta p_v}$$

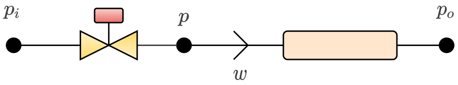
    where w_v is the inlet to outlet mass flowrate, ρ the liquid density, Δp_v the inlet minus outlet pressure drop; $x \in [0,1]$ is the valve opening and $\Phi(x)$ the intrinsic characteristic, with $\Phi(0) = 0$ and $\Phi(1) = 1$; C_v is the flow coefficient;
  - the hydraulic load (l subscript) is ruled by

$$\Delta p_l = K_l \rho w_l^2$$

    with the same notation and conventions; K_l is a known parameter, we assume unidirectional flow (reasonable) and neglect gravity for simplicity.

Case 1

Assignment 1 — sizing C_v



- We assume a nominal flowrate w_n is required, hence in nominal (sizing) conditions

$$p = p_o + K_l \rho w_n^2.$$

- As such with valve fully open we get

$$C_v \sqrt{\rho \left(p_i - (p_o + K_l \rho w_n^2) \right)} = w_n.$$

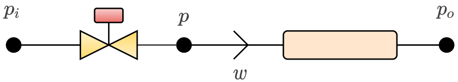
- Solving for C_v yields the nominal value for C_v as

$$C_{vn} = \frac{w_n}{\sqrt{\rho (p_i - p_o - K_l \rho w_n^2)}}.$$

- In general we then take $C_v = \alpha C_{vn}$, with $\alpha > 1$ (not too much).

Case 1

Assignment 2 — installed characteristic



- Definition: variable to govern expressed as a function of the valve opening.
- In our case, $w(x)$ or $p(x)$ depending on the purpose of the control system.
- In the w case we start from $w = C_v\Phi(x)\sqrt{\rho(p_i - p_o - K_l\rho w^2)}$, and solving for w (recall the assumed signs) we obtain

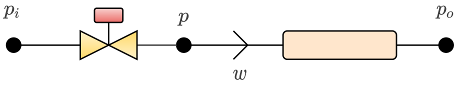
$$w(x) = \frac{C_v\Phi(x)\sqrt{\rho(p_i - p_o)}}{\sqrt{1 + K_l\rho^2 C_v^2\Phi^2(x)}}.$$

- In the p case we just need to bring in $p(x) = p_o + K_l\rho w^2(x)$, whence

$$p(x) = p_o + \frac{K_l C_v^2\Phi^2(x)\rho^2(p_i - p_o)}{1 + K_l\rho^2 C_v^2\Phi^2(x)}.$$

Case 1

Assignment 3 — linearisation, physical version



- Purpose: select $\Phi(x)$ so that the installed characteristic be linear (or affine) in x.
- We address the w case. With the selected C_v (no matter α) setting $x=1$ gives

$$w_{max} = \frac{C_v \sqrt{\rho(p_i - p_o)}}{\sqrt{1 + K_l \rho^2 C_v^2}}.$$

while obviously $x=0$ gives $w=0$.

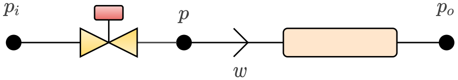
- We want to determine $\Phi(x)$ such that

$$w(x) = w_{max} \cdot x$$

i.e., in this case, a truly linear installed characteristic.

Case 1

Assignment 3 — linearisation, physical version



- We rewrite $w(x) = w_{max} \cdot x$ expressing $w(x)$ and w_{max}, that is

$$\frac{C_v \Phi(x) \sqrt{\rho (p_i - p_o)}}{\sqrt{1 + K_l \rho^2 C_v^2 \Phi^2(x)}} = \frac{C_v \sqrt{\rho (p_i - p_o)}}{\sqrt{1 + K_l \rho^2 C_v^2}} x,$$

  and solve for $\Phi(x)$, which gives

$$\Phi(x) = \frac{x}{\sqrt{1 + K_l \rho^2 C_v^2 (1 - x^2)}}.$$

- Now, we just need to find a valve with this intrinsic characteristic in some supplier catalogue ☺

Case 1

Assignment 3 — linearisation, physical version

- First, however, let us consider for simplicity

$$\Phi(x) = \frac{x}{\sqrt{1+\beta(1-x^2)}}$$

and take its inverse

$$x(\Phi) = \frac{\Phi\sqrt{1+\beta}}{\sqrt{1+\beta\Phi^2}}.$$

- Note the similarity of $x(\Phi)$ to the expression we derived before for $w(x)$: $x(\Phi)$ in fact represents the *shape* of the installed flow characteristic if the intrinsic one were linear.

- The question is: how relevant is the above nonlinearity for control? And as a consequence, how precisely does one need to linearise?

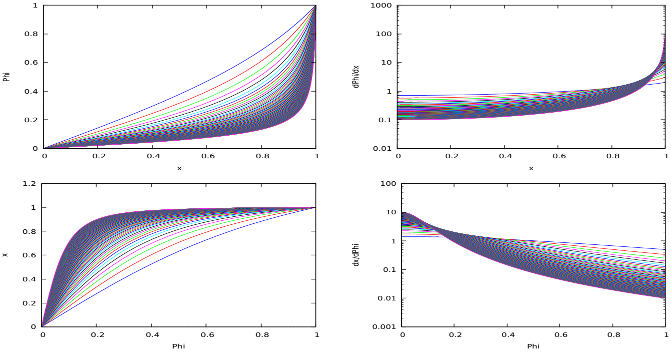
Case 1

Assignment 3 — linearisation, physical version

- wxMaxima:

```
Phix  : makelist(x/sqrt(1+beta*(1-x^2)),beta,1,100); xPhi  : makelist(Phi*sqrt((beta+1)/(beta*Phi^2+1)),beta,1,100);
plot2d(Phix,[x,0,1],[xlabel,"x"],[ylabel,"Phi"],[legend,false]);
plot2d(diff(Phix,x),[x,0,1],[xlabel,"x"],[logy],[ylabel,"dPhi/dx"],[legend,false]);
plot2d(xPhi,[Phi,0,1],[xlabel,"Phi"],[ylabel,"x"],[legend,false]);
plot2d(diff(xPhi,Phi),[Phi,0,1],[xlabel,"Phi"],[logy],[ylabel,"dx/dPhi"],[legend,false]);
```

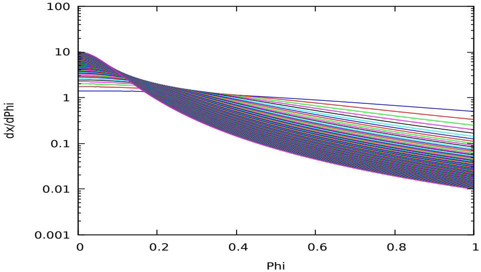
- Note: we make the $d\Phi/dx$ and $dx/d\Phi$ axes logarithmic for better readability.



Case 1

Assignment 3 — linearisation, physical version

- Let us concentrate on $dx/d\Phi$:



- Depending on how relevant the denominator is wrt the (linear) numerator, the differential gain change seen by a flow controller can range from a factor 4–5 up to 3 orders of magnitude.
- However things are better if we focus on the typical opening range of a valve for smooth operation, say 0.2–0.8.

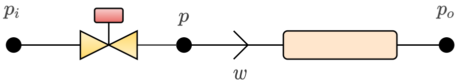
Case 1

Assignment 3 — first lessons learnt (and software moving in)

- As the actuator dynamics is low-order and normally decently damped, achieving good gain margins is generally no big deal.
- Thus, once linearisation reduces the intrinsic valve nonlinearity to an $x \rightarrow w$ gain variation within a factor well below the gain margin in the range of interest, its duty can be taken for accomplished.
- If necessary to achieve tight control, the "rest" of the linearisation can be done in software, by cascading a nonlinear compensating characteristic to the controller (a task once devoted to mechanical cams).
- Methodological hint for the enthusiast: absolute stability.
- Anyway, this is why on the market one finds valves with standard intrinsic characteristics (linear, equi-percent, quick-opening...) designed to fit most needs. Throw a glance at some catalogue on the net if curious, there are many.

Case 1

Assignment 3 — back to linearisation, physical and/or software (now we know)



- Now (briefly) for the p case. For this installed characteristic we got

$$p(x) = p_o + \frac{K_l C_v^2 \Phi^2(x) \rho^2 (p_i - p_o)}{1 + K_l \rho^2 C_v^2 \Phi^2(x)}.$$

- Setting $x = 0$ and $x = 1$ yields respectively

$$p_{x0} = p_o, \qquad p_{x1} = p_o + \frac{K_l C_v^2 \rho^2 (p_i - p_o)}{1 + K_l \rho^2 C_v^2}.$$

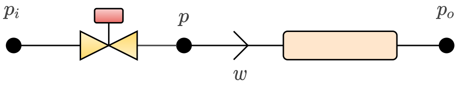
- We want to determine $\Phi(x)$ such that

$$p(x) = p_{x0} + (p_{x1} - p_{x0}) \cdot x$$

i.e., in this case, an *affine* installed characteristic.

Case 1

Assignment 3 — linearisation



- We rewrite $p(x) = p_{x0} + (p_{x1} - p_{x0}) \cdot x$ expressing $p(x)$, p_{x0} and p_{x1}, that is

$$p_o + \frac{K_l C_v^2 \Phi^2(x) \rho^2 (p_i - p_o)}{1 + K_l \rho^2 C_v^2 \Phi^2(x)} = p_o + \frac{K_l C_v^2 \rho^2 (p_i - p_o)}{1 + K_l \rho^2 C_v^2} x,$$

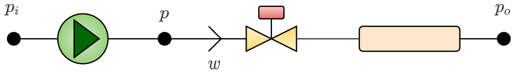
and solve for $\Phi(x)$, which gives

$$\Phi(x) = \sqrt{\frac{x}{1 + K_l \rho^2 C_v^2 (1 - x)}}.$$

- Analogous considerations as for the w case apply.

Case 2

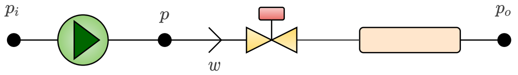
Setting



- Assignment:
  1. size the valve flow coefficient;
  2. determine the valve installed characteristic;
  3. make the said characteristic linear.
- Definitely quite similar to Case 1: for compactness and clarity we concentrate on the essentials.

Case 2

Setting



- Hypotheses:
  - p_i and p_o are prescribed boundary conditions;
  - valve and hydraulic load are ruled respectively by

    $$w_v = C_v \phi(x) \sqrt{\rho \Delta p_v}, \qquad \Delta p_l = K_l \rho w_l^2$$

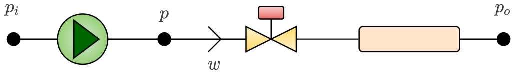
    with the same notation and conventions we used for Case 1;
  - the pump (subscript p) is centrifugal, fixed-speed and ruled by

    $$\Delta p_p = H_0 - H_1 w_p^2$$

    where H_0 and H_1 are known positive parameters while (beware) Δp_p equals the *output* minus the *input* pressure; we still assume flows to be unidirectional and neglect gravity effects.

Case 2

Assignment 1 — sizing C_v



- From the component constitutive laws and the connections we have

$$p = p_i + H_0 - H_1 w^2, \qquad w = C_v \Phi(x) \sqrt{\rho \left(p - (p_o + K_l \rho w^2)\right)}$$

  whence

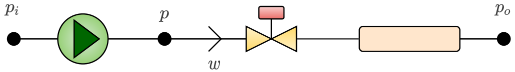
$$w = C_v \Phi(x) \sqrt{\rho \left(p_i - p_o + H_0 - (H_1 + K_l \rho) w^2\right)}$$

- Requiring a nominal flowrate ($w = w_n$) with valve fully open ($\Phi = 1$) we thus get the nominal C_v as

$$C_{vn} = \frac{w_n}{\sqrt{\rho \left(p_i - p_o + H_0 - (H_1 + K_l \rho) w^2\right)}}.$$

Case 2

Assignment 2 — installed characteristic



- We see only the w case. Solving

$$w = C_v\Phi(x)\sqrt{\rho\left(p_i - p_o + H_0 - (H_1 + K_l\rho)w^2\right)}$$

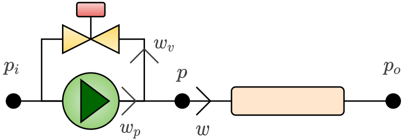
for w we get

$$w(x) = \frac{C_v\Phi(x)\sqrt{\rho\left(p_i - p_o + H_0\right)}}{\sqrt{1 + C_v^2\rho\Phi^2(x)(H_1 + K_l\rho)}}.$$

- Very similar to Case 1; assignment 3 left as an exercise.
- Suggested question: what happens as H_1 approaches zero? Why?

Case 3

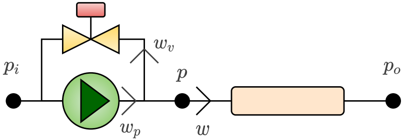
Setting



- Assignment:
  1. size the valve flow coefficient;
  2. determine the valve installed characteristic;
  3. make the said characteristic linear.
- This is a bit different from Cases 1 and 2: again, we concentrate on the essentials.

Case 3

Setting



- Hypotheses:
  - p_i and p_o are prescribed boundary conditions;
  - valve and hydraulic load (same conventions as above) are ruled respectively by

$$w_v = C_v \phi(x) \sqrt{\rho \Delta p_v}, \qquad \Delta p_l = K_l \rho w_l^2.$$

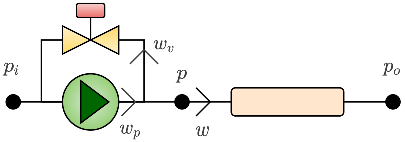
  - the pump (subscript p) is fixed-speed and volumetric (typical choice), hence ruled by

$$w_p = w_{p0}.$$

  - We still assume flows to be unidirectional, and neglect gravity effects.

Case 3

Assignment 1 — sizing C_v



- From the component constitutive laws and the connections we have

$$p = p_o + K_l \rho w^2, \quad w = w_{p0} - w_v, \quad w_v = C_v \Phi(x) \sqrt{\rho (p - p_i)},$$

  whence

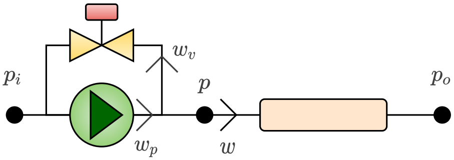
$$w = w_{p0} - C_v \Phi(x) \sqrt{\rho (p_o - p_i + K_l \rho w^2)}$$

- Requiring a nominal flowrate ($w = w_n$) with valve fully open ($\Phi = 1$) we get the nominal C_v as

$$C_{vn} = \frac{w_{p0} - w_n}{\sqrt{\rho (p_o - p_i + K_l \rho w_n^2)}}.$$

Case 3

Assignment 2 — installed characteristic



- We see again only the w case.
- Solving

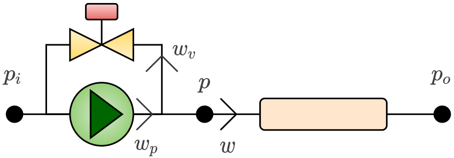
$$w = w_{p0} - C_v \Phi(x) \sqrt{\rho \left(p_o - p_i + K_l \rho w^2\right)}$$

for w, we get

$$w(x) = \sqrt{\frac{w_{p0}^2 - 2C_v \sqrt{\rho \left(p_o - p_i + K_l \rho w^2\right)} w_{p0} \Phi(x) + C_v^2 \rho \left(p_o - p_i\right) \Phi^2(x)}{1 - K_l \rho^2 C_v^2 \Phi^2(x)}}.$$

Case 3

Assignment 2 — installed characteristic



- Substituting 0 and 1 for x we respectively have

$$w_{x0} = w_{p0}, \quad w_{x1} = \sqrt{\frac{w_{p0}^2 - 2C_v\sqrt{\rho(p_o - p_i + K_l\rho w^2)}w_{p0} + C_v^2\rho(p_o - p_i)}{1 - K_l\rho^2C_v^2}}.$$

- Intuitively, and subject to the obvious physical feasibility conditions we omitted throughout for brevity, w_{x1} must be less than w_{x0}.
- Hence in this case we have an affine installed characteristic with a negative slope in x.
- We do not discuss further for space reasons.

Flow/pressure control with liquids

Wrap-up

- The scheme is composed of pumps, valves, piping (and tanks).
- One actuates with combinations of pumps and valves (we saw examples).
- In closed circuits $p_i = p_o$ and there is typically some pressure reference (such as vents to atmosphere, large tanks or pressurisers), hence our examples apply.
- Sizing and control synthesis are intertwined owing to inherent nonlinearity.
- Dynamics are of low order, concentrated in the actuators, while differential gains can vary a lot.
- The typical control structure is a dynamic controller, most often of PI/PID type, with possibly a static output characteristic if physical linearisation is not enough.
- Once linearised, controller tuning is straightforward.

Foreword

- We stick to liquids as thermovectors of election.
- Our point is to control temperature at some point in a pipe (e.g., the outlet of a heating station).
- We have two major ways to do that:
  1. release thermal power to the pipe upstream the controlled point (generally to heat);
  2. mix in another liquid stream at different temperature (generally to cool).
- We shall see when dealing with heat networks that the two actions may need combining — as one can already guess, incidentally.
- Action 1 is quite straightforward to implement.
- Action 2 is conversely a bit more tricky owing to sensor dynamics; we are now investigating this.

Controlled system

- Let w_p be the incoming "process" flowrate at temperature T_p, to join with the "control" flowrate w_c at temperature T_c so as to get a mix temperature T°.
- Assume T_c to be ideally controlled — for simplicity; even if not so the main point in the following treatise holds.
- Assume that w_c obeys to a control input u through a dominantly 1st-order dynamics (reasonable in the case u acts on a valve positioner, possibly with an inner flow loop).
- Assuming instantaneous mixing (reasonable in turbulent flows), the resulting temperature T – also interpretable as the equilibrium one with constant inputs – is

$$T = \frac{w_p T_p + w_c T_c}{w_p + w_c}.$$

Controlled system

- Linearising and assuming all inputs but w_c to not vary for brevity, we have

$$\delta T = \mu \delta w_c, \qquad \mu = \frac{\overline{w}_p(\overline{T}_c - \overline{T}_p)}{(\overline{w}_p + \overline{w}_c)^2}.$$

  where bars denote equilibrium values.

- Assuming now $u \in [0,1]$ and denoting by $w_{c,max}$ and τ_c the maximum w_c and the $u \rightarrow w_c$ time constant we obtain

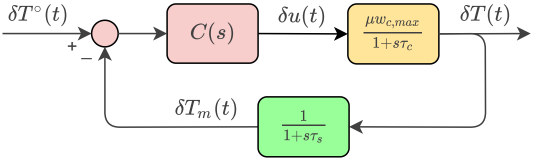
$$\delta T = \frac{\mu w_{c,max}}{1+s\tau_c}\delta u$$

- and finally, denoting by τ_s the time constant of the required T sensor we can express the measured mixing temperature (variation) as

$$\delta T_m = \frac{\mu w_{c,max}}{(1+s\tau_c)(1+s\tau_s)}\delta u.$$

Control block diagram

- Indicating by $C(s)$ the controller for δT acting on δu leads to

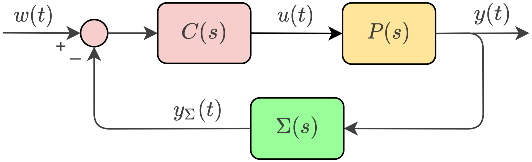


- Problem: there are cases where τ_s is remarkably larger than τ_c.
- As a result, achieving a good control of δT_m in general does not mean controlling well δT, which is the real objective.

- WARNING: an autotuning controller (often used for mass tuning of "simple" loops) would exactly aim for a good control of δT_m: any "auto-" or "self-something" object needs instructing... ☺

Control block diagram

- Let us study the problem with a lighter notation for simplicity:



- We have

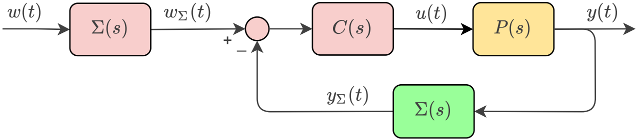
$$\frac{Y(s)}{W(s)} = \frac{C(s)P(s)}{1+C(s)P(s)\Sigma(s)}$$

so that aiming for $Y(s)/W(s) = F(s)$ results in

$$C(s) = \frac{F(s)}{P(s)(1-\Sigma(s)F(s))}.$$

Control synthesis

- Idea: require y_Σ to track not w but instead w filtered through Σ, *i.e.*,



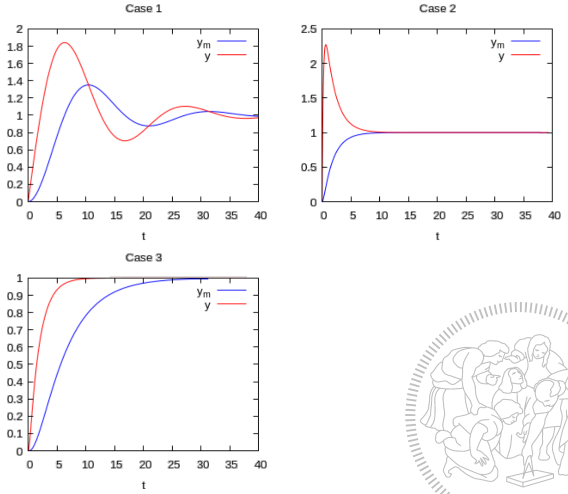
  where the controller includes a sensor *replica*.
- This is sensible, as if y_Σ tracks the output of Σ then y will track its input, that is w...
  ...as long as Σ *replica* in the controller matches the physical Σ in the process (closely enough in the band of interest).
- Doing so, one can tune $C(s)$ by shaping the loop frequency response $C(j\omega)P(j\omega)\Sigma(j\omega)$ as usual.

Example

— try changing Ts/Tp and ω_c, making the set point filter `Sf` (slightly) differ from sensor `S`,...

wxMaxima:

```
data : [mu=1,Tp=1,Ts=5,wc=0.5];

P    : mu/(1+s*Tp);
S    : 1/(1+s*Ts);

C1   : wc/s/P;                 /* PI            */
C2   : wc/s/P/S/(1+s/(10*wc)); /* Real PID      */
Sf   : S;                      /* S filter for w  */

/* Closed-loop responses of y_m and y to a w step */

yc1  : ilt(subst(data,C1*P/(1+C1*P*S))/s,s,t);
yc2  : ilt(subst(data,C2*P/(1+C2*P*S))/s,s,t);
yc3  : ilt(subst(data,Sf*C2*P/(1+C2*P*S))/s,s,t);

ymc1 : ilt(subst(data,C1*P*S/(1+C1*P*S))/s,s,t);
ymc2 : ilt(subst(data,C2*P*S/(1+C2*P*S))/s,s,t);
ymc3 : ilt(subst(data,Sf*C2*P*S/(1+C2*P*S))/s,s,t);

wxplot2d([ymc1,yc1],[t,0,40],[legend,"y_m","y"]);
wxplot2d([ymc2,yc2],[t,0,40],[legend,"y_m","y"]);
wxplot2d([ymc3,yc3],[t,0,40],[legend,"y_m","y"]);
```



Wrap-up, remarks, lesson learnt

- Clearly, whatever choice for `Sf`, preserve the unit gain (no need to explain why).
- We did not investigate the idea of tuning a feedback C by cancelling the (dominant) dynamics of S instead of $P \Rightarrow$ exercise.

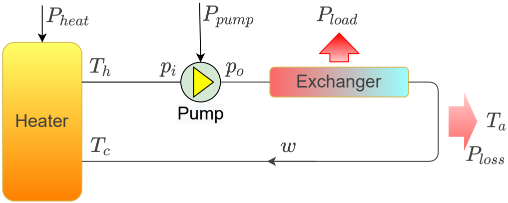
- But anyway, is this matter energy-relevant?
- It can be. Overtemperatures (transiently) increase losses, see for example $y(t)$ in case 2 of the previous example.

- In any case, most of the times the sensor dynamics is not the main point, however as we just saw this is not completely general. Keep in mind.

Foreword

- Suppose you have to provide a certain heat rate to a body or ambient by means of a thermovector fluid (here, liquid).
- This means in general that you need to govern
  - a temperature or a temperature difference across some exchanger,
  - and possibly a fluid flowrate.
- Are the two controls independent?
- Which are the energy-relevant aspects of the problem?
- Let us investigate.

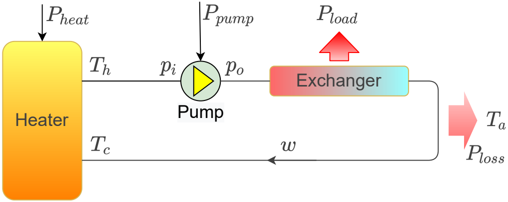
Controlled system



- Assuming no pump heating nor gravity effects (for simplicity and without impairing the following treatise) the heating and pumping powers are respectively

$$
\begin{aligned}
P_{heat} &= wc(T_h - T_c) \\
P_{pump} &= w\,\frac{p_o - p_i}{\rho} = w\,K_{piping}w^2 = K_{piping}w^3
\end{aligned}
$$

Controlled system



- As for the loss power, we simplistically – yet here adequately – refer it to the average fluid temperature, and assume a constant-conductance exchange with an exogenous ambient temperature T_a; this yields

$$P_{loss} = G\left(\frac{T_h + T_c}{2} - T_a\right).$$

Control requirements

- Let us assume to control w and T_h.
- This can be accomplished with two independent loops quite straightforwardly.
- But how to select the set points w°, T_h°?
- A reasonable criterion is to minimise the total power $P_{tot} = P_{heat} + P_{pump} + P_{loss}$, that is, given the load to fulfil,

$$\min_{w^\circ, T_h^\circ} P_{tot} \qquad s.t.\ P_{heat} = P_{load} + P_{loss}.$$

- As in T-project years the KKT equations (already known at this point in E-project years) come later, we exploit the presence of only two variables.
- Just recall – in particular if we had more decision variables – that this is clearly constrained optimisation.

Set point determination

- wxMaxima:

```
e1      : Ploss = G*((Th+Tc)/2-Ta);
e2      : Pheat = Pload+Ploss;
e3      : Pheat = w*c*(Th-Tc);
s1      : solve([e1,e2,e3],[Ploss,Th,Pheat]);
sTh     : rhs(s1[1][2]);
sPheat  : rhs(s1[1][3]);
Ppump   : Kpiping*w^3;
Ptot    : subst(s1,Pheat+Ppump);
```

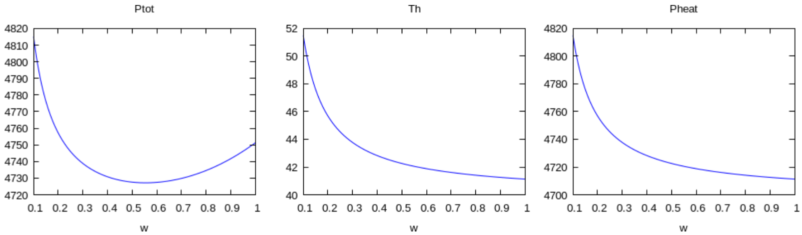
- We obtain (reintroducing the set point mark omitted in the script)

$$P_{tot} = K_{piping} w^{\circ 3} + \frac{2w^{\circ} c \left(P_{load} + G(T_c - T_a)\right)}{2w^{\circ} c - G}.$$

Set point determination

Example 1 — just to give an idea of the orders of magnitude

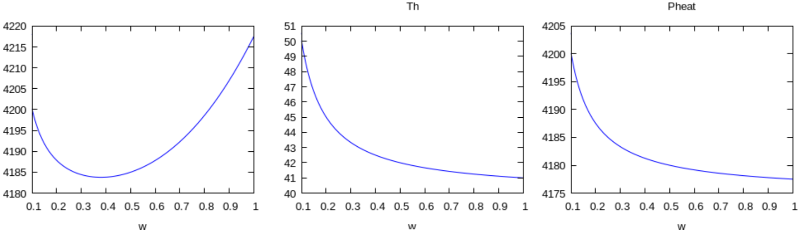
- Small residential case:
  - pressure drop $1m_{H_2O} \approx 10^4 Pa$ for a nominal flowrate of $0.5 kg/s$ and $\rho = 1000 kg/m^3$ $\Rightarrow K_{piping} = 40$;
  - $c = 4186 J/kg°C$, $P_{load} = 4000W$, $T_c = 40°C$ (must ensure a minimum fluid T);
  - $T_a = 5°C$ (winter), $G = 20W/°C$ (quite poor insulation).
- Result:



Set point determination

Example 2 — same purpose

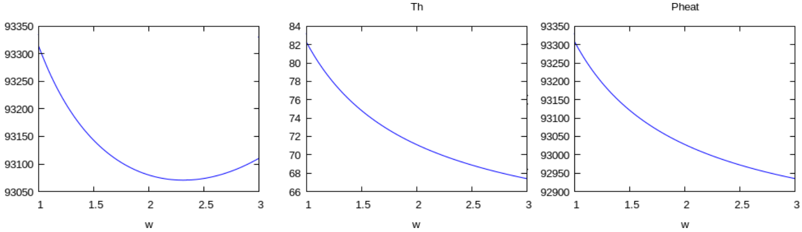
- Same as example 1 but with $G = 5W/°C$ (quite good insulation).
- Result:



Set point determination

Example 3 — same purpose

- Slightly larger case:
  - pressure drop $5m_{H_2O} \approx 0.5bar$ for a nominal flowrate of $10m^3/h \approx 2.8kg/s$ and $\rho = 1000kg/m^3 \Rightarrow K_{piping} = 6.4$;
  - $c = 4186J/kg°C$, $P_{load} = 90kW$, $T_c = 60°C$;
  - $T_a = 5°C$ (winter), $G = 50W/°C$ (good insulation).
- Result:



Wrap-up and lessons learnt

- We do not delve into mathematics to determine the optimum set points, as the assumptions we made are fine for evidencing the issue to face but not that much for precise computations (you have a library with realistic models for that).
- However:
  - though often w cannot be modulated and T_h° is computed based on ambient conditions with "climatic curves", if w control is available joint set point optimisation can be an option;
  - with liquid thermovectors P_{pump} should be far smaller (hardly ever above 1–2% or so) of P_{heat}, but in general pumps are electric, and an electric kWh (to date) costs more than a thermal one.
- In any case, when the objective to attain comes from the combined effect of two or more set points, and for these there are multiple choices that for the said objective are in fact equivalent, at least consider the possibility of introducing some set point optimisation.

Thermal systems – control problems (part 2)

Thermal control with central/local sources

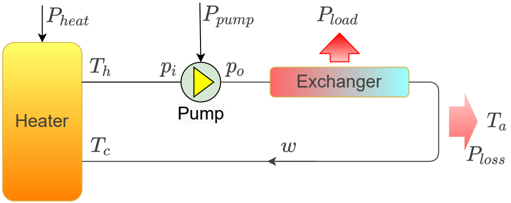
Thermal control with actuator equalisation

Keeping (comfort) variables within limits w/ overrides

Foreword

- Suppose you have to provide a certain heat rate to a body or ambient by means of a thermovector fluid (here, liquid).
- This means in general that you need to govern
  - a temperature or a temperature difference across some exchanger,
  - and possibly a fluid flowrate.
- Are the two controls independent?
- Which are the energy-relevant aspects of the problem?
- Let us investigate.

Controlled system



- Assuming no pump heating nor gravity effects (for simplicity and without impairing the following treatise) the heating and pumping powers are respectively

$$
\begin{aligned}
P_{heat} &= wc(T_h - T_c) \\
P_{pump} &= w\,\frac{p_o - p_i}{\rho} = w\,K_{piping}w^2 = K_{piping}w^3
\end{aligned}
$$

Thank you for your attention

alberto.leva@polimi.it