Optimal Control problems

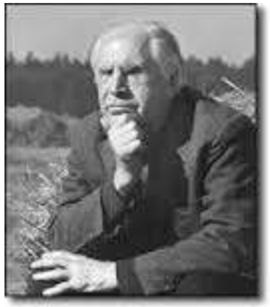
- CV focus on a "curve"
- There is no plant dynamics
- There is no manipulated variable

Furthermore:

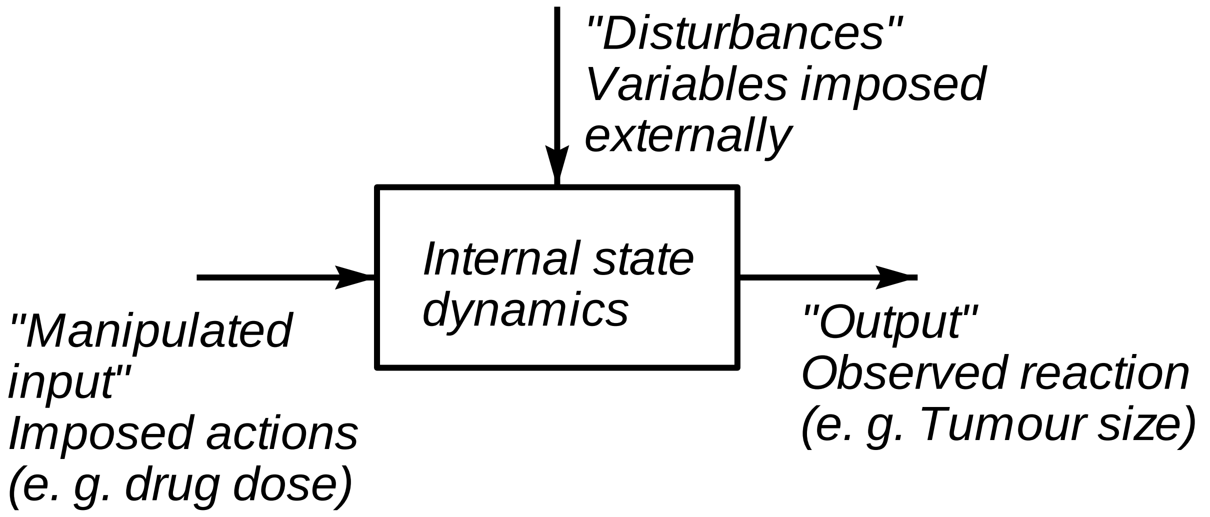
- There is the need to consider (at least) Piecewise C^1 functions (*e. g*., bang-bang control)

Need for:

- A new problem structuring (Optimal Control)
- Another tool: Pontryagin's Maximum Principle (1956).



Example: Control approach to therapy design in cancer



Compute the therapy along time that yields the best compromise between

- maximizing the therapeutic effect (minimize tumor size)
- minimizing a measure of toxic effects (minimize total treatment)

Controlled Gompertz model

$$\dot{x} = \alpha\, x\, log\left(\frac{M}{x}\right) - \beta\, x\, u$$

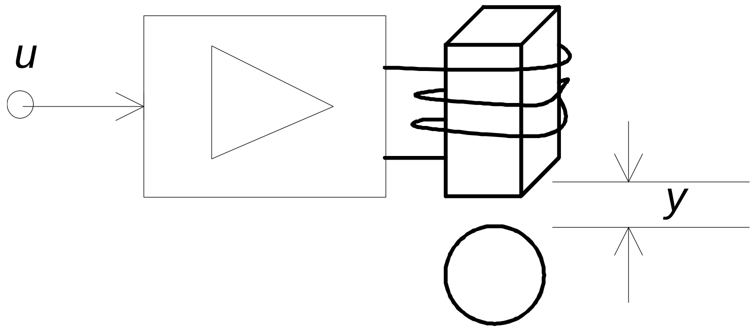
Associated control problem

$$J_2 = x(T) + \rho \int_0^T u(t)\, dt$$

$$0 \leq u \leq u_{max}$$

A detour: The state model

Examplo: A simple magnetic suspension



Newton's law:

$$m\frac{d^2y}{dt^2} = \sum applied \;\; forces \qquad \rightarrow \qquad \frac{d^2y}{dt^2} = u$$

The model is described by a 2nd order differential equation.

Take as state variables position and velocity

$$x_1(t) = y(t) \qquad x_2(t) = \dot{y}(t)$$

With these variables, the system is described by 2 first order differential equations (instead of 1 2nd order differential equation)

$$\begin{cases} \dfrac{dx_1}{dt} = x_2 \\ \dfrac{dx_2}{dt} = u \end{cases}$$

These equations form the state model of the magnetic suspension and x_1 and x_2 are the state variables that define the state space.

The system

$$\begin{cases} \dfrac{dx_1}{dt} = x_2 \\ \dfrac{dx_2}{dt} = u \end{cases}$$

may be written in the equivalent matrix for

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u \qquad y = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

The vector $\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$ is the state vector of the system.

Definition of state

The state of a system is a set of variables such that, if you know them at a time instant, and the inputs that act on the system from that instant on, then you can compute the state for all future instants.

The state at an instance summarizes all relevant information to compute future states. There is no need to know how the state from which you star was attained.

The set of the first order differential equations verified by the state vector (as a function of time) is the state model.

Standard form of the state equations (linear case)

In the case of the magnetic suspension

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u \qquad y = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

Define the matrices

$$A = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \qquad B = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \qquad C = \begin{bmatrix} 1 & 0 \end{bmatrix} \qquad D = 0$$

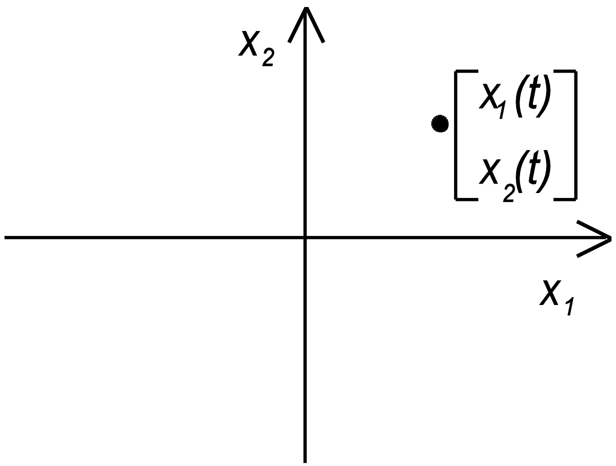
The state model is written in the standard form as

$$\dot{x} = Ax + bu$$

$$y = Cx + Du$$

The state space of the magnetic suspension

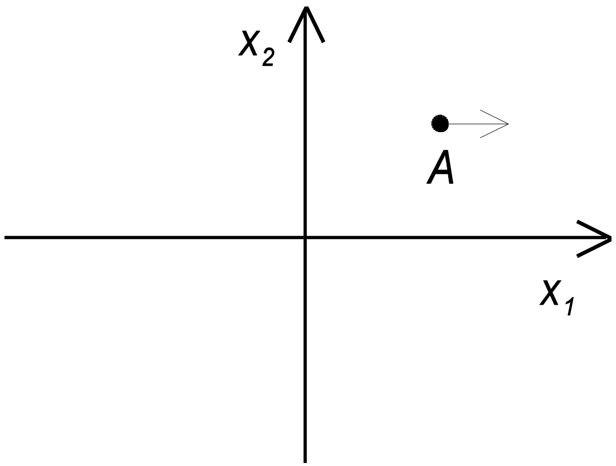
A major advantage of the state model is that we can imagine state evolution in geometrical terms in the space defined by $\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$.



This space is the state space.

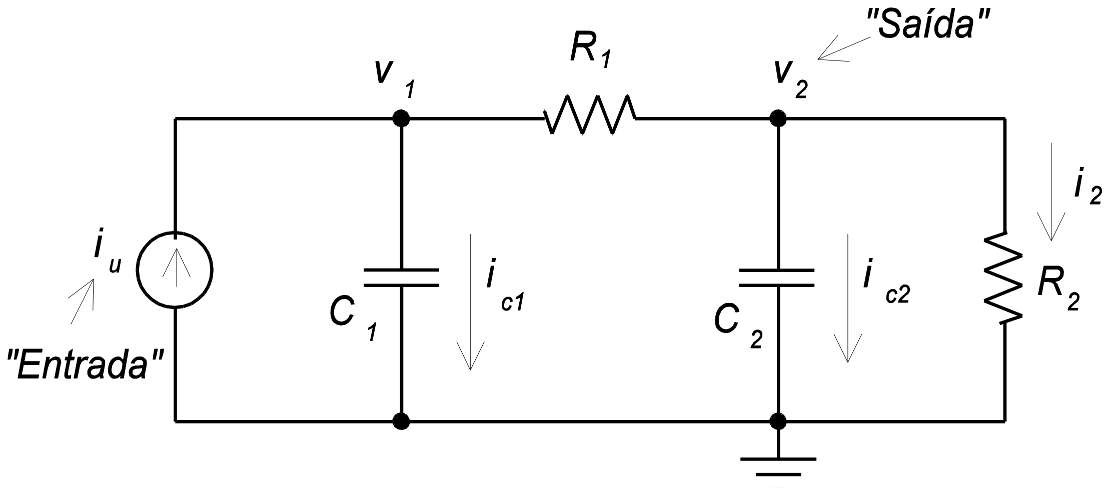
When time passes, the evolution of the variables corresponds to a trajectory of $\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$ on state-space.

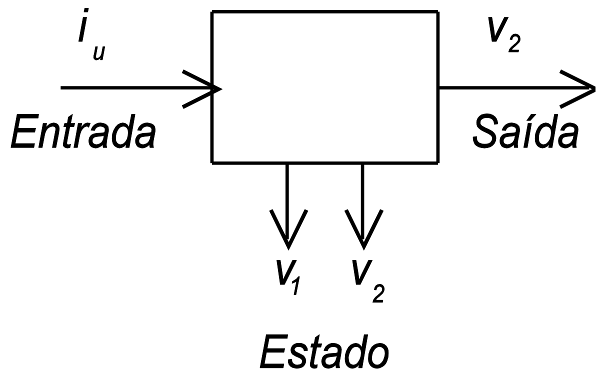
For instance, if the sphere is at point A in state-space, it will move according to the arrow (*explain why*).



Exercise: Consider different initial states in each of the four quadrants. In each case, sketch the state trajectories. Plot the corresponding evolutions of x_1 and x_2 against time.

Example: Electrical circuit





$$\begin{cases} C_1 \dfrac{dv_1}{dt} = i_u - \dfrac{v_1 - v_2}{R_1} \\ C_2 \dfrac{dv_2}{dt} = \dfrac{v_1 - v_2}{R_1} - \dfrac{v_2}{R_2} \end{cases}$$

$$
\begin{cases}
C_1 \dfrac{dv_1}{dt} = i_u - \dfrac{v_1 - v_2}{R_1} \\
C_2 \dfrac{dv_2}{dt} = \dfrac{v_1 - v_2}{R_1} - \dfrac{v_2}{R_2}
\end{cases}
$$

Define the matrices

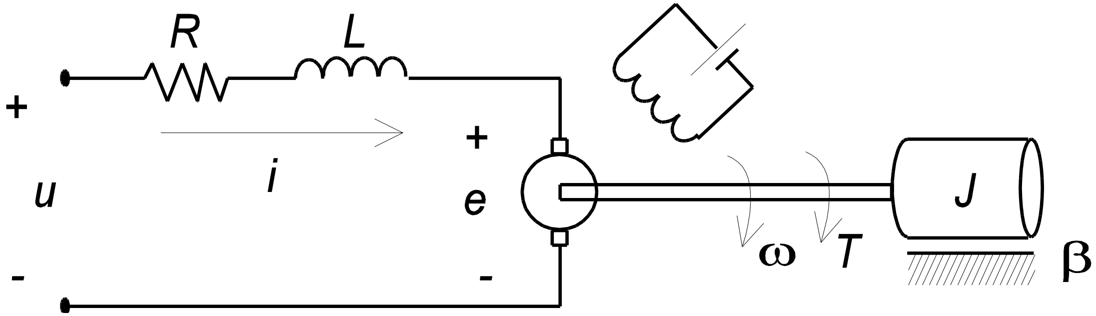
$$
A = \begin{bmatrix} -\dfrac{1}{C_1 R_1} & -\dfrac{1}{C_1 R_1} \\ \dfrac{1}{C_2 R_1} & -\dfrac{1}{C_2 R_1} - \dfrac{1}{C_2 R_2} \end{bmatrix} \quad B = \begin{bmatrix} 1/C_1 \\ 0 \end{bmatrix} \quad C = \begin{bmatrix} 0 & 1 \end{bmatrix} \quad D = 0 \quad u = i_u \quad x = \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} \quad y = v_2
$$

Again, the standard form of the state model of the electrical circuit is

$$
\dot{x} = Ax + Bu
$$

$$
y = Cx + Du
$$

Still another example: Direct current motor model with armature control



Motor torque

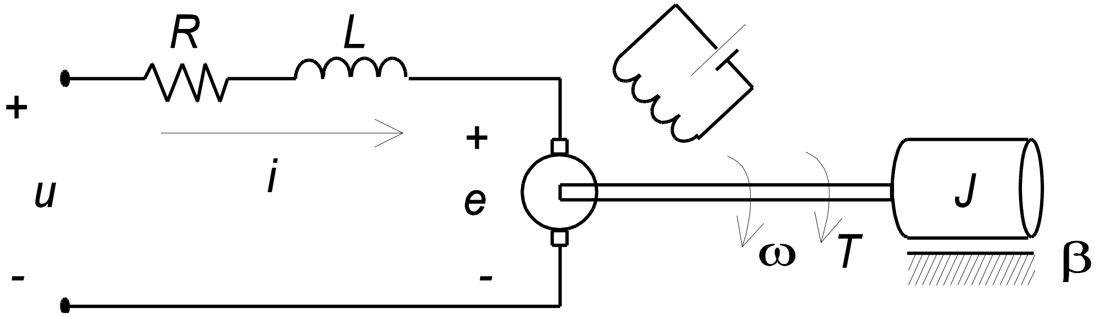
$$T(t) = K'\phi(t)\, i(t)$$

Since the magnetic flux ϕ created by the armature is constant,

$$T(t) = Ki(t)$$

Electrical tension induced at the rotor terminals

$$e = K_b \omega$$



Circuit of the motor rotor:

$$L\frac{di}{dt} + R\,i + e = u \quad \text{or} \quad L\frac{di}{dt} + R\,i + K_b\omega = u$$

Movimento do rótor do motor:

$$J\frac{d\omega}{dt} = T(t) - \beta\omega \quad \text{or} \quad J\frac{d\omega}{dt} = Ki - \beta\omega$$

Take as state variables

$$x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} \omega \\ i \end{bmatrix}$$

State equations with output ω:

$$\dot{x} = \begin{bmatrix} -\frac{\beta}{J} & \frac{K}{J} \\ -\frac{K_b}{L} & -\frac{R}{L} \end{bmatrix} x + \begin{bmatrix} 0 \\ \frac{1}{L} \end{bmatrix} u$$

$$y = \begin{bmatrix} 1 & 0 \end{bmatrix} x$$

Exercise: Modify this model to include as a state the angular position θ.

State model (linear case)

State equation (differential equation, relates the input u with the state x):

$$\dot{x}(t) = Ax(t) + Bu(t)$$

State initial condition

$$x(0) = x_0$$

Output equation (algebraic equation, relates the state x with the output y):

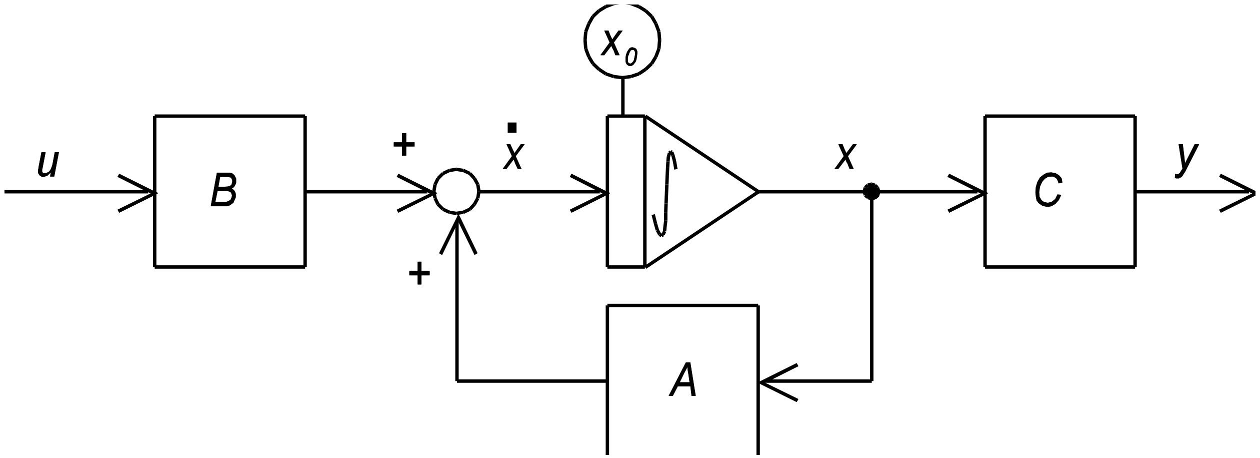
$$y(t) = Cx(t) + Du(t)$$

Dimensions:

$$x(t) \in R^n, \quad u(t) \in R^m, \quad y(t) \in R^p \qquad A[n \times n] \quad B[n \times m] \quad C[p \times n] \quad D[p \times m]$$

Usually $D = 0, \quad m = 1, \quad p = 1$. (scalar system with more poles than zeros)

Block diagram of the linear state model



$$\dot{x}(t) = Ax(t) + Bu(t)$$

Exercise: Learn how to use SIMULINK and simulate the state models of the previous examples. In the case of the DC motor, add a proportional feedback controller to control the angular position.

Choice of the state variables

The state variables are not unique.

- Physical variables
  - Positions of the parts that move independently and their derivatives
  - Tensions across capacitors and currents across coils
  - Variables associated with energy storage, including entropy
- Mathematical state (no physical meaning, see below).
  - Phase variables (the output and its n-1 derivatives for systems of order n without zeros).
  - …

Computing the transfer function from the state model

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

Take the Laplace trsnform with zero initial conditions:

$$sX(s) = AX(s) + bU(s)$$
$$Y(s) = CX(s)$$

$$X(s) = TL(x) \quad U(s) = TL(u)$$

From here

$$(sI - A)X(s) = bU(s) \quad \longrightarrow \quad X(s) = (sI - A)^{-1}bU(s)$$

or

$$Y(s) = C(sI - A)^{-1}b\,U(s)$$

Transfer function from the state model

$$G(s) = C(sI - A)^{-1} b$$

Since

$$(sI - A)^{-1} = \frac{adj(sI - A)}{\det(sI - A)}$$

The transfer function is

$$G(s) = \frac{C\, adj(sI - A) b}{\det(sI - A)}$$

The MATLAB function *ss2tf* solves this problem.

Note on Linear Algebra – Adjoint of a matrix

The adjoint of a matrix $M = \left[m_{ij} \right]$ is

$$adj(M) = \left[M_{ij} \right]^T$$

where M_{ij} is the co-factor of the matrix entry m_{ij}, that is, is the determinant of the matrix that results by eliminating from M line i and column j, multiplied by -1^{i+j}.

Example: $$adj\left(\begin{bmatrix} a & b \\ c & d \end{bmatrix} \right) = \begin{bmatrix} d & -b \\ -c & a \end{bmatrix}$$

Adjoint of a matrix – Example

$$M = \begin{bmatrix} 1 & 2 & 3 \\ 5 & 0 & 0 \\ 0 & 6 & 1 \end{bmatrix} \qquad adj(M) = \begin{bmatrix} 0 & -5 & 30 \\ 16 & 1 & -6 \\ 0 & 15 & -10 \end{bmatrix}^T = \begin{bmatrix} 0 & 16 & 0 \\ -5 & 1 & 15 \\ 30 & -6 & -10 \end{bmatrix}$$

To check this result, observe that

$$M\,\frac{adj(M)}{\det(M)} = \frac{1}{80}\begin{bmatrix} 1 & 2 & 3 \\ 5 & 0 & 0 \\ 0 & 6 & 1 \end{bmatrix}\begin{bmatrix} 0 & 16 & 0 \\ -5 & 1 & 15 \\ 30 & -6 & -10 \end{bmatrix} = I_3$$

Reference: G. Strang, *Linear Algebra and its Applications*, 2ª ed., p 170.

System poles and zeros

$$G(s) = \frac{C\,adj(sI - A)b}{\det(sI - A)}$$

The characteristic polynomial of matrix A is

$$\det(sI - A)$$

The poles are the roots of the characteristic polynomial of matrix A.

The zeros are the roots of the polynomial

$$C\,adj(sI - A)b$$

Transfer function from the state model – Example

$$A = \begin{bmatrix} -5 & -6 \\ 1 & 0 \end{bmatrix} \quad b = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \quad C = \begin{bmatrix} 0 & 1 \end{bmatrix}$$

$$sI - A = \begin{bmatrix} s+5 & 6 \\ -1 & s \end{bmatrix} \qquad (sI - A)^{-1} = \frac{1}{s(s+5)+6} \begin{bmatrix} s & -6 \\ 1 & s+5 \end{bmatrix}$$

$$G(s) = \frac{1}{s(s+5)+6} \begin{bmatrix} 0 & 1 \end{bmatrix} \begin{bmatrix} s & -6 \\ 1 & s+5 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \frac{1}{(s+2)(s+3)}$$

Getting a state model from the TF – Systems without zeros

Given the transfer function without zeros

$$G(s) = \frac{b_0}{s^3 + a_1 s^2 + a_2 s + a_3}$$

We want to compute a state model that represents it.

This state model is not unique.

State variables: For a system of order n, the output and its first $n-1$ derivatives. In this example, $n = 3$.

From the transfer function

$$G(s) = \frac{b_0}{s^3 + a_1 s^2 + a_2 s + a_3} ,$$

start by obtaining the differential equation:

$$s^3 Y(s) + a_1 s^2 Y(s) + a_2 s Y(s) + a_3 Y(s) = b_0 U(s)$$

From here we get the differential equation

$$\dddot{y}(t) + a_1 \ddot{y}(t) + a_2 \dot{y}(t) + a_3 y(t) = b_0 u(t)$$

$$\dddot{y}(t)+a_1\ddot{y}(t)+a_2\dot{y}(t)+a_3y(t)=b_0u(t)$$

State variables (output and its derivatives up to order $n-1=2$):

$$x_1=y$$
$$x_2=\dot{y}=\dot{x}_1$$
$$x_3=\ddot{y}=\dot{x}_2$$

From the differential equation

$$\dot{x}_3=-a_1x_3-a_2x_2-a_3x_1+b_0u(t)$$

State model:

$$\dot{x}_1 = x_2$$

$$\dot{x}_2 = x_3$$

$$\dot{x}_3 = -a_1 x_3 - a_2 x_2 - a_3 x_1 + b_0 u(t)$$

In matrix notation:

$$\dot{x} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -a_3 & -a_2 & -a_1 \end{bmatrix} x + \begin{bmatrix} 0 \\ 0 \\ b_0 \end{bmatrix} u$$

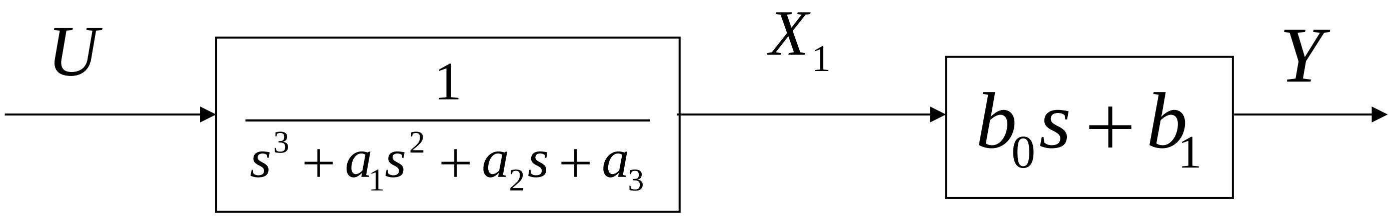
$$y = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} x$$

Systemas with zeros

$$G(s) = \frac{b_0 s + b_1}{s^3 + a_1 s^2 + a_2 s + a_3}$$

The previous technique may not be applied with changes because it leads to a derivative of the input.

One possibility (there are other!) is to "split" the transfer function into its zeros and poles, taking as state variables the output with poles and its first $n-1$ derivatives



The equation of dynamic is the same as before.

The output equation is changed:

$$y = b_0 \dot{x}_1 + b_1 x_1 = b_0 x_2 + b_1 x_1$$

$$y = \begin{bmatrix} b_1 & b_0 & 0 \end{bmatrix} x$$

Coordinate transform

Consider the model

$$\dot{x}(t) = Ax(t) + bu(t)$$

$$y(t) = Cx(t)$$

Perform a coordinate transform

$$z(t) = Tx(t)$$

where T is a square matrix that is invertible.

What is the state model verified by vector $z(t)$?

Suggestion: Differentiate $z(t) = Tx(t)$

$$z(t) = Tx(t)$$

Differentiate to get

$$\dot{z}(t) = T\dot{x}(t)$$

Use the state model of $x(t)$:

$$\dot{z}(t) = T(Ax(t) + bu(t))$$

Use the inverse transform:

$$\dot{z}(t) = TAT^{-1}z(t) + Tbu(t))$$

$$y(t) = Cx(t) = CT^{-1}x(t)$$

Conclusion: Change of coordinates in the state model

Given the state model

$$\dot{x}(t) = Ax(t) + bu(t) \qquad\qquad y(t) = Cx(t)$$

we do a gtransfer of coordinates

$$z(t) = Tx(t)$$

where T is a square and invertible matrix.

In the new coordinates, the state equations are

$$\dot{z}(t) = \mathrm{E}z(t) + \Gamma u(t) \qquad\qquad y(t) = Hx(t)$$

$$E = TAT^{-1} \qquad\qquad H = CT^{-1}$$

Orbits and time functions

Example:

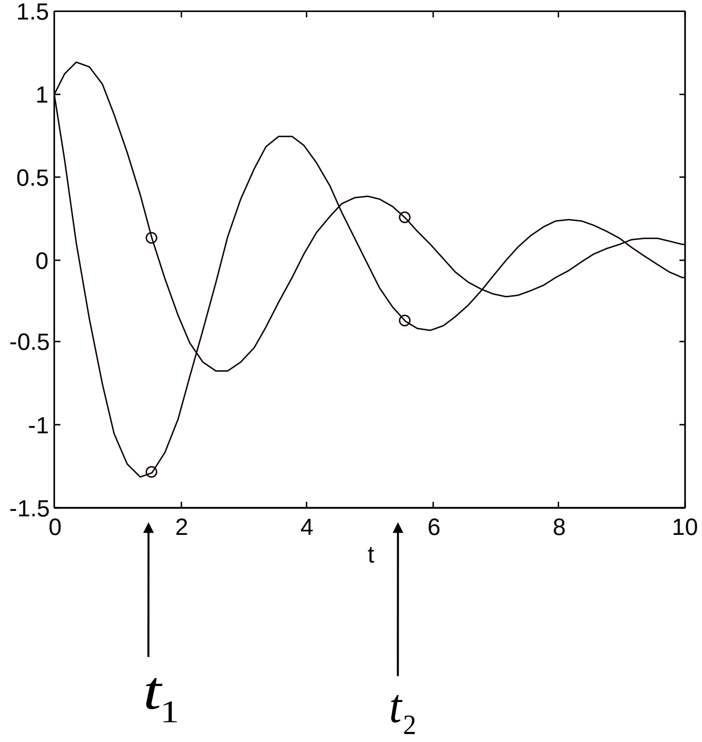
$$\frac{dx_1}{dt} = x_2$$

$$\frac{dx_2}{dt} = -2x_1 - 2x_2$$

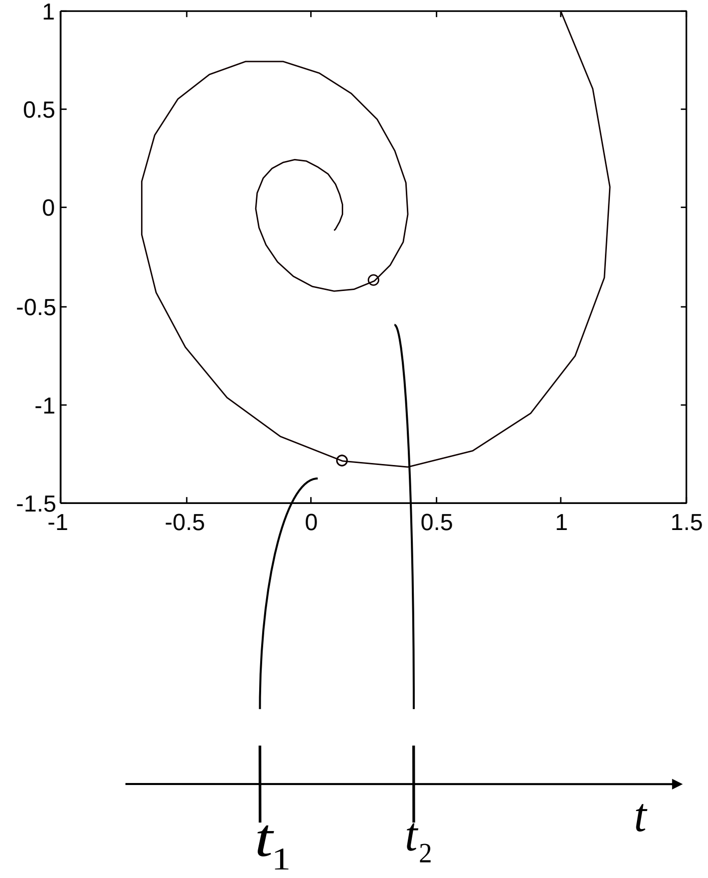
With initial condition $x_1(0)=1 \quad x_2(0)=1$. The orbit (trajectory in state-space) that starts from this initial condition is obtained by eliminating the time from the two time functions $x_1(t)$ and $x_2(t)$.

Exercise: Simulate this example in SIMULINK/MATLAB. Consider different initial equations and draw the orbits in state-space superimposed.

Time response of the state variables



Corresponding orbit



We can add more orbits, corresponding to other initial conditions

The homogeneous equation and the orbits

Homogeneous equation:

$$\dot{x}(t) = Ax(t)$$

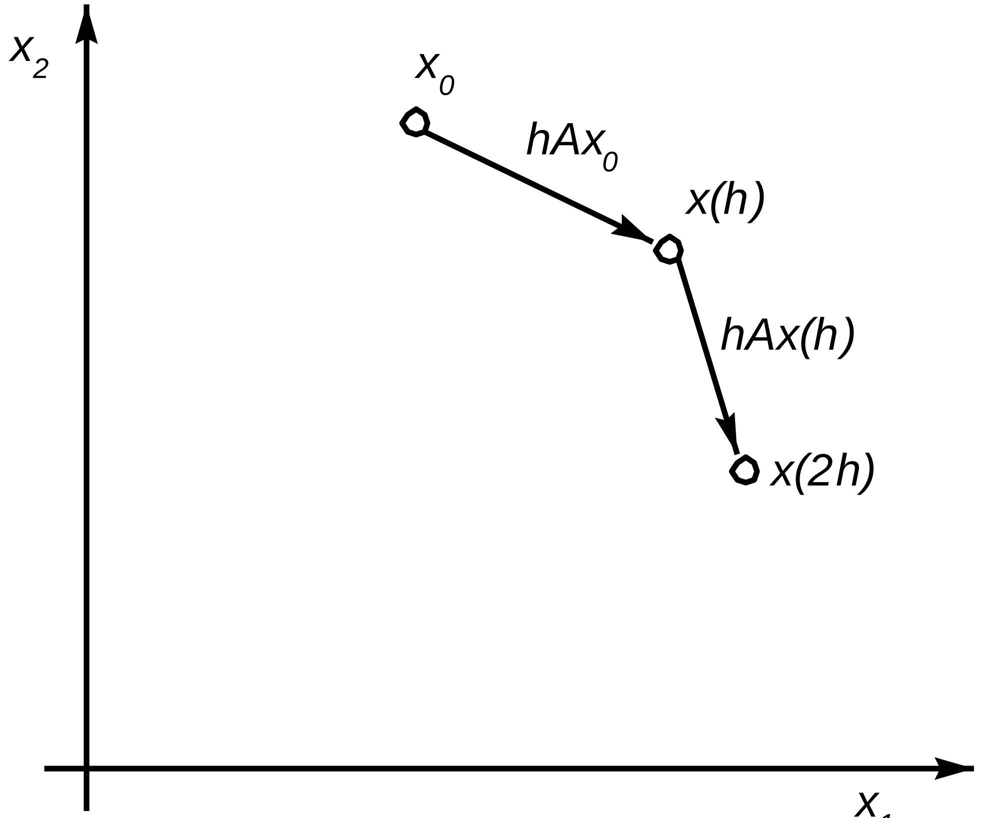
Approximating the derivative by finite differences:

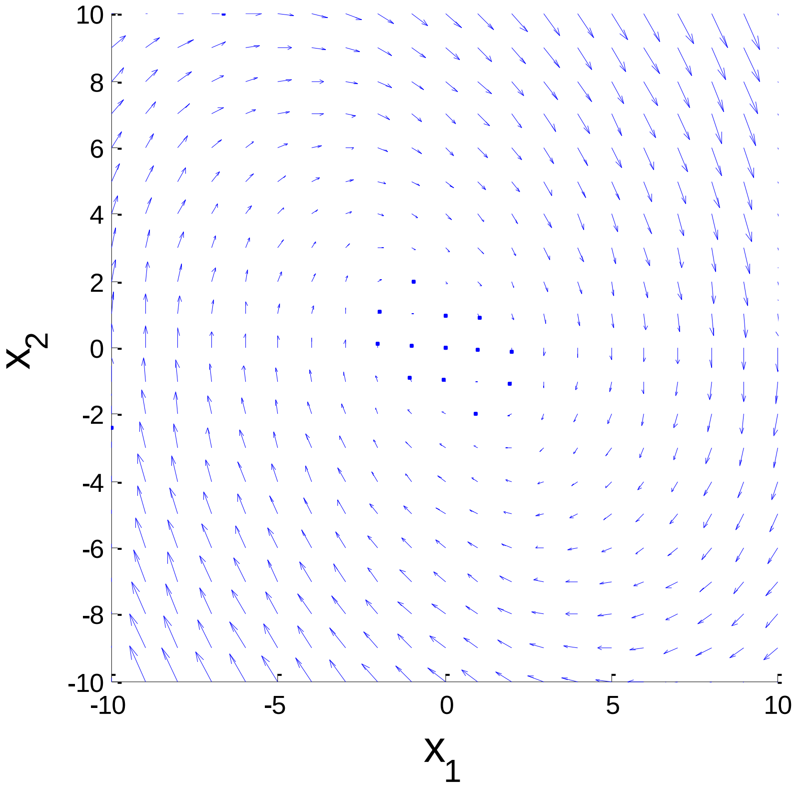
$$\dot{x}(t) \approx \frac{x((k+1)h) - x(kh)}{h}$$

The state homogeneous equation may be approximated by the difference equation

$$x((k+1)h) = x(kh) + h\,Ax(kh)$$

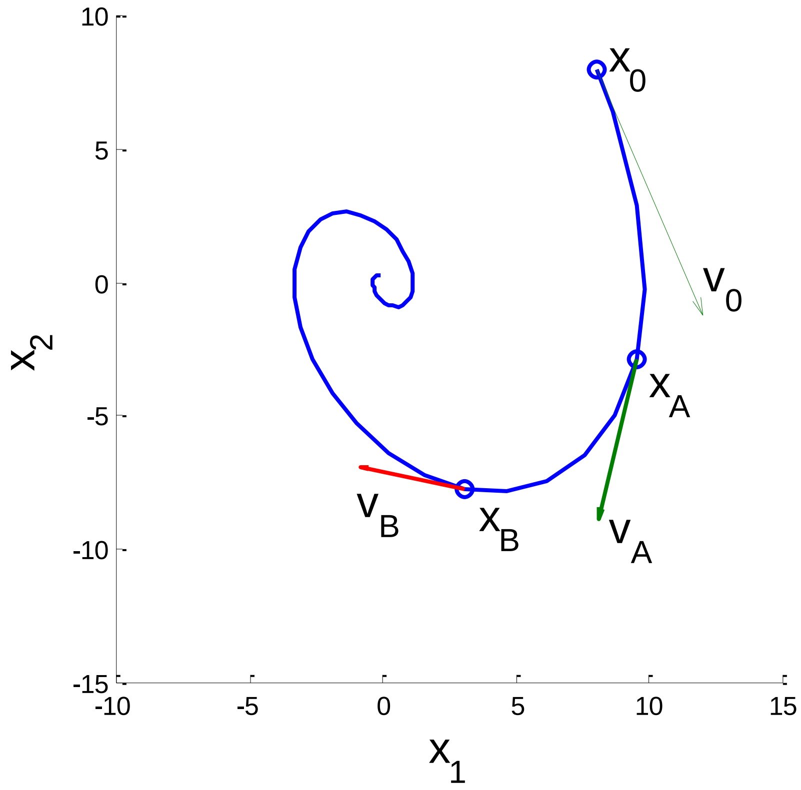
$$x((k+1)h) = x(kh) + h\,Ax(kh)$$





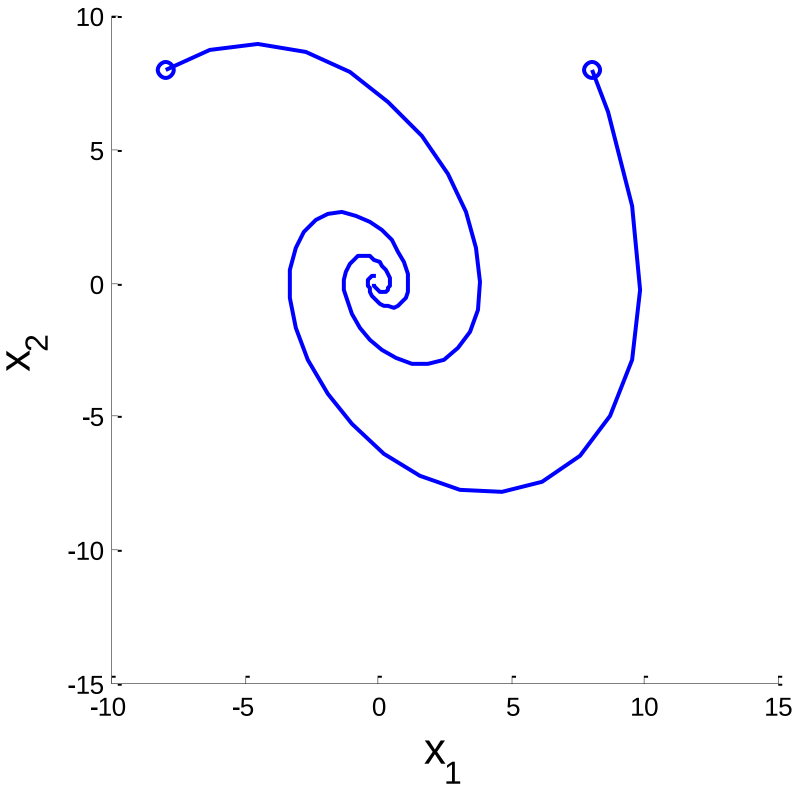
At each point x of state space, the function Ax defines a vector (vector field) that points the direction to be followed at that point by the orbit.

The **vector field** may be plotted in MATLAB using the function *quiver*.

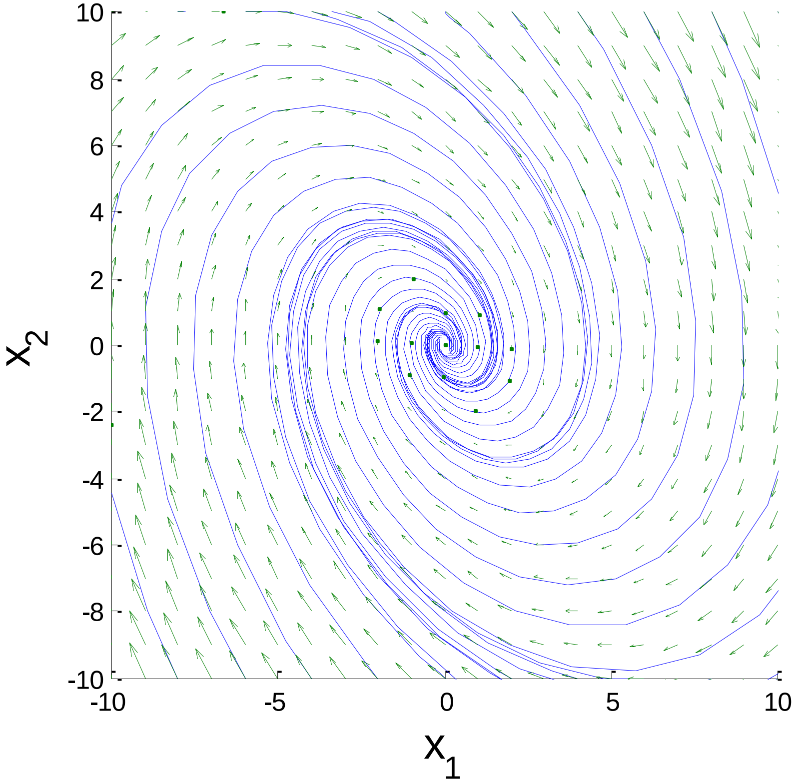


Starting at x_0, the solution advances locally in the direction $v_0 = Ax_0$.

At each point the orbit is tangent to the vector field.



If we start at another initial condition, we get another orbit. The figure shows two of such orbits.



Starting from different initial conditions (there are infinite ones!) we get the **phase portrait**.

Note on Linear Algebra: Eigenvalues and eigenvectors

Given a square matrix A $[n \times n]$, the eigenvectors v_i satisfy

$$Av_i = \lambda_i v_i$$

where λ_i is the corresponding eigenvalue.

At most, there are n linearly independ eigenvectors (but there can be less).

The eigenvectors have also the name of *mode vectors*.

Computing eigenvalues and eigenvectors

Since

$$Av_i = \lambda_i v_i$$

The eigenvectors satisfy the set of algebraic equations

$$(A - \lambda_i I)v_i = 0$$

In order for this set to have non-trivial solutions $v_i \neq 0$, the eigenvalues λ_i must satisfy the polynomial equation

$$\det(A - \lambda_i I) = 0$$

To compute the eigenvalues and eigenvectors of a square matrix A, proceed thus as follows:

a) Compute the eigenvalues by solving the polynomial equation

$$\det(A - \lambda_i I) = 0$$

b) For each eigenvalue λ_i obtain the corresponding eigenvector by solving the system of algebraic equations

$$(A - \lambda_i I)v_i = 0$$

Since this system is undetermined, its solution is obtained up to a normalizing constant, that is chosen in the most convenient way.

MATLAB function: *eig*

Computing eigenvalues and eigenvectors

$$A = \begin{bmatrix} 4 & -5 \\ 2 & -3 \end{bmatrix} \qquad A - \lambda I = \begin{bmatrix} 4-\lambda & -5 \\ 2 & -3-\lambda \end{bmatrix}$$

Chatacteristic polynomial:

$$\det(A - \lambda I) = 0$$

$$\begin{vmatrix} 4-\lambda & -5 \\ 2 & -3-\lambda \end{vmatrix} = (4-\lambda)(-3-\lambda) + 10 = \lambda^2 - \lambda - 2 = (\lambda+1)(\lambda-2)$$

The eigenvalues are the roots of this polynomial:

$$\lambda_1 = -1 \qquad \lambda_2 = 2$$

Eigenvectors:

$$\lambda_1 = -1 \qquad (A - \lambda_1 I)v_1 = \begin{bmatrix} 5 & -5 \\ 2 & -2 \end{bmatrix} \begin{bmatrix} v_{1,1} \\ v_{1,2} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

The solution is any multiple of $v_1 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$

$$\lambda_2 = 2 \qquad (A - \lambda_2 I)v_2 = \begin{bmatrix} 2 & -5 \\ 2 & -5 \end{bmatrix} \begin{bmatrix} v_{2,1} \\ v_{2,2} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

The solution is any multiple of $v_2 = \begin{bmatrix} 5 \\ 2 \end{bmatrix}$

Matrix diagonalization

Assumption: The matrix A has n lineraly independent eigenvectors.

Modal matrix (the columns are the eigenvectors of A):

$$M = \begin{bmatrix} v^1 & \cdots & v^n \end{bmatrix}$$

Diagonal matrix of eigenvalues

$$\Lambda = diag(\lambda_1, \ldots, \lambda_n)$$

Warning: Not all matrices verify this assumption.

Since, for each eigenvector

$$Av_i = \lambda_i v_i$$

It is

$$AM = M\Lambda$$

or:

$$A = M\Lambda M^{-1}$$

Or else, multiplying at the right by M and at the left by M^{-1}:

$$\Lambda = M^{-1}AM$$

Solving the homogeneous equation by diagonalization

Valid when A has n linearly independent eigenvectors.

$$\dot{x}(t) = Ax(t) \qquad x(0) = x_0$$

Change of coordinates associated to the modal matrix:

$$z = M^{-1}x \quad \text{ou} \quad x = Mz$$

In the z coordinates, the dynamics becomes

$$\dot{z} = M^{-1}\dot{x} = M^{-1}Ax = M^{-1}AMz = \Lambda z$$

Thus, the components of z are decoupled, and the equations may be solved separately!

$$\dot{z} = \Lambda z$$

This matrix equation corresponds to the set of 1st order differential equations

$$\begin{cases} \dot{z}_1 = \lambda_1 z_1 \\ \vdots \\ \dot{z}_n = \lambda_n z_n \end{cases}$$

Since the equations are decoupled, they may be solved separately:

$$z_1(t) = k_1 e^{\lambda_1 t}$$

$$\ldots$$

$$z_n(t) = k_n e^{\lambda_n t}$$

The k_i are constants that depend on the initial conditions.

Structure of the response in the x coordinates:

$$x = Mz = \begin{bmatrix} v_1 & \cdots & v_n \end{bmatrix} \begin{bmatrix} k_1 e^{\lambda_1 t} \\ \vdots \\ k_2 e^{\lambda_2 t} \end{bmatrix}$$

or

$$x = k_1 v_1 e^{\lambda_1 t} + \ldots + k_n v_n e^{\lambda_n t}$$

Each of the terms

$$v_i e^{\lambda_i t}$$

Is called a system mode. The response is thus a linear combination of the modes, in which the coefficients depend on the initial conditions.

Example

The time response of the homogeneous system

$$\dot{x}(t) = Ax(t)$$

with

$$A = \begin{bmatrix} 4 & -5 \\ 2 & -3 \end{bmatrix} \qquad x(0) = \begin{bmatrix} 8 \\ 5 \end{bmatrix}$$

has the form:

$$x(t) = k_1 \begin{bmatrix} 1 \\ 1 \end{bmatrix} e^{-1t} + k_2 \begin{bmatrix} 5 \\ 2 \end{bmatrix} e^{2t}$$

Computing k_1 and k_2 from the initial conditions:

$$x(t) = k_1 \begin{bmatrix} 1 \\ 1 \end{bmatrix} e^{-1t} + k_2 \begin{bmatrix} 5 \\ 2 \end{bmatrix} e^{2t}$$

For $t = 0$:

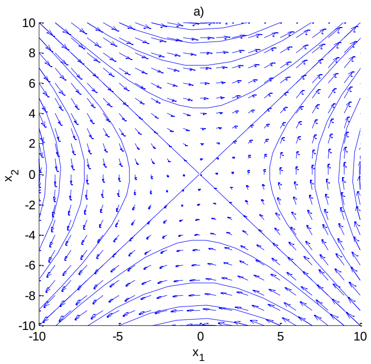
$$\begin{bmatrix} 8 \\ 5 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \end{bmatrix} k_1 + \begin{bmatrix} 5 \\ 2 \end{bmatrix} k_2$$

This system of algebraic equations may be written as

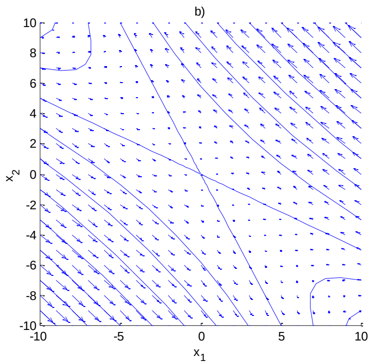
$$\begin{bmatrix} 1 & 5 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} k_1 \\ k_2 \end{bmatrix} = \begin{bmatrix} 8 \\ 5 \end{bmatrix} \quad \rightarrow \quad k_1 = 3, \quad k_2 = 1$$

Exercise

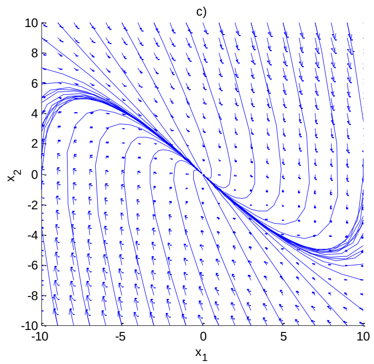
A)



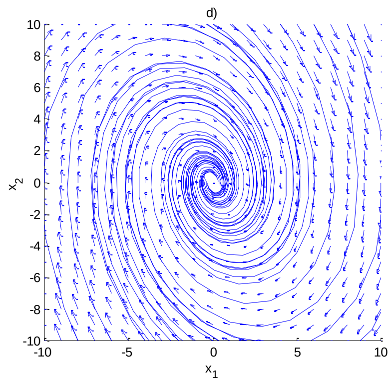
B)



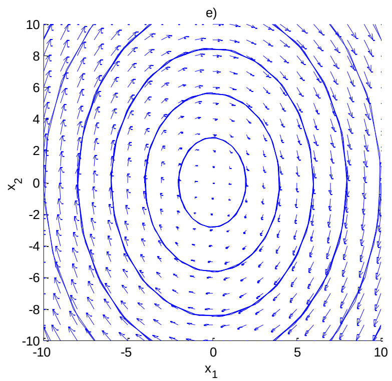
C)



D)



E)



$$A_1 = \begin{bmatrix} 0 & 1 \\ -2 & -0.6 \end{bmatrix} \qquad A_2 = \begin{bmatrix} 0 & 1 \\ -2 & -3 \end{bmatrix} \qquad A_3 = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

$$A_4 = \begin{bmatrix} 0 & 1 \\ -2 & 0 \end{bmatrix} \qquad A_5 = \begin{bmatrix} -5/3 & -4/3 \\ 4/3 & 5/3 \end{bmatrix}$$

Match the matrices with the phase portraits.

Peano-Baker series and the state transition matrix

$$\dot{x} = Ax \qquad x(t_0) = x_0$$

Has the solution

$$x(t) = \Phi(t, t_0) x(t_0)$$

The matrix $\Phi(t, t_0)$ is called the state transition matrix and for time invariant systems is given by the Peano-Baker series that defines the matrix exponential

$$\Phi(t, t_0) = e^{A(t-t_0)} = I + A(t-t_0) + \frac{1}{2!}A^2(t-t_0)^2 + \frac{1}{3!}A^3(t-t_0)^3 + \ldots$$

This series is different for the time varying case.

Time invariant systems: Computing the state transition matrix with the Laplace transform

$$\dot{x} = Ax \qquad x(0) = x_0$$

Take the Laplace transform:

$$sX - x_0 = AX$$

$$X = (sI - A)^{-1} x_0$$

$$x(t) = TL^{-1}\left\{(sI - A)^{-1}\right\} x_0$$

Conclusion:

$$\Phi(t, t_0) = TL^{-1}\left\{(sI - A)^{-1}\right\}$$

Example: Computing the state transition matrix with the Laplace transf.

Consider

$$A = \begin{bmatrix} 1 & 1 \\ 4 & 1 \end{bmatrix}$$

Compute the state transition matrix using the Laplace transform.

Solution:

$$\Phi(t, t_0) = TL^{-1}\left\{(sI - A)^{-1}\right\}$$

$$sI - A = \begin{bmatrix} s-1 & -1 \\ -4 & s-1 \end{bmatrix} \qquad \det(sI - A) = (s-3)(s+1)$$

$$(sI-A)^{-1}=\begin{bmatrix}\dfrac{s-1}{(s-3)(s+1)} & \dfrac{1}{(s-3)(s+1)}\\ \dfrac{4}{(s-3)(s+1)} & \dfrac{s-1}{(s-3)(s+1)}\end{bmatrix}$$

$$\frac{s-1}{(s-3)(s+1)}=\frac{A}{s-3}+\frac{B}{s+1}=\frac{1}{2}\left(\frac{1}{s-3}+\frac{1}{s+1}\right)\qquad A=\frac{3-1}{3+1}=\frac{1}{2}\qquad B=\frac{-2}{-4}=\frac{1}{2}$$

$$\phi_{11}(t)=\phi_{22}(t)=\frac{1}{2}\left(e^{3t}+e^{-t}\right)$$

$$\frac{1}{(s-3)(s+1)}=\frac{A}{s-3}+\frac{B}{s+1}=\frac{1}{4}\left(\frac{1}{s-3}-\frac{1}{s+1}\right)\qquad A=\frac{1}{4}\qquad B=-\frac{1}{4}$$

$$\phi_{12}(t)=\frac{1}{4}\left(e^{3t}-e^{-t}\right)\qquad \phi_{21}(t)=e^{3t}-e^{-t}$$

$$\Phi(t,0)=\begin{bmatrix} \frac{1}{2}\left(e^{3t}+e^{-t}\right) & \frac{1}{4}\left(e^{3t}-e^{-t}\right) \\ e^{3t}-e^{-t} & \frac{1}{2}\left(e^{3t}+e^{-t}\right) \end{bmatrix}$$

End of example

Invertibility of the state transition matrix

Abel-Jacobi-Liouville theorem (special case for time invariant systems):

$$\det\left[e^{A(t-t_0)}\right] = e^{(t-t_0)trA}$$

Where the *trace of* A, denoted by trA, is the sum of the diagonal elements.

This theorem allows to conclude that the state transition matrix is always invertible because the real exponential never vanishes.

This property is also true in the time varying case.

Non-homogeneous (forced) systems

$$\dot{x}(t) = Ax(t) + bu(t)$$

Solution

$$x(t) = e^{A(t-t_0)}x_0 + \int_{t_0}^{t} e^{A(t-\tau)}bu(\tau)d\tau$$

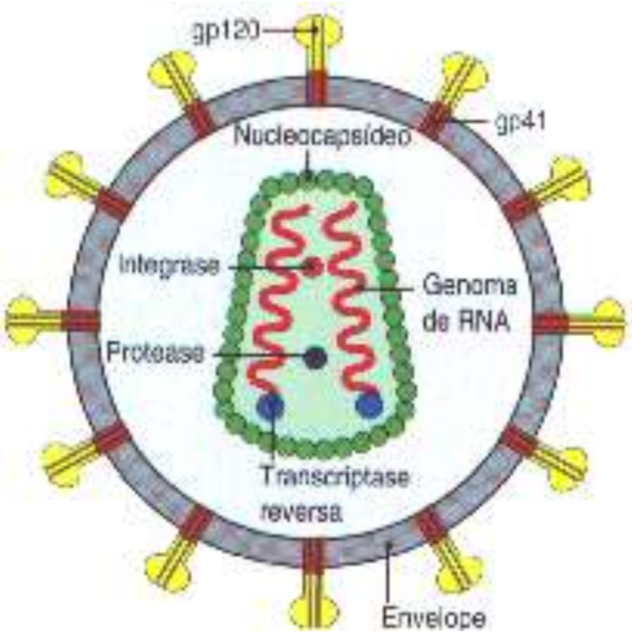
Free regime

Forced regime

Methods of proof:

- Direct verification (se Leibniz's rule to differentiate an integral)
- Principle of superposition
- Change of variable (justifies the name *Variation of constants formula*)

Nonlinear systems. HIV-1 infection

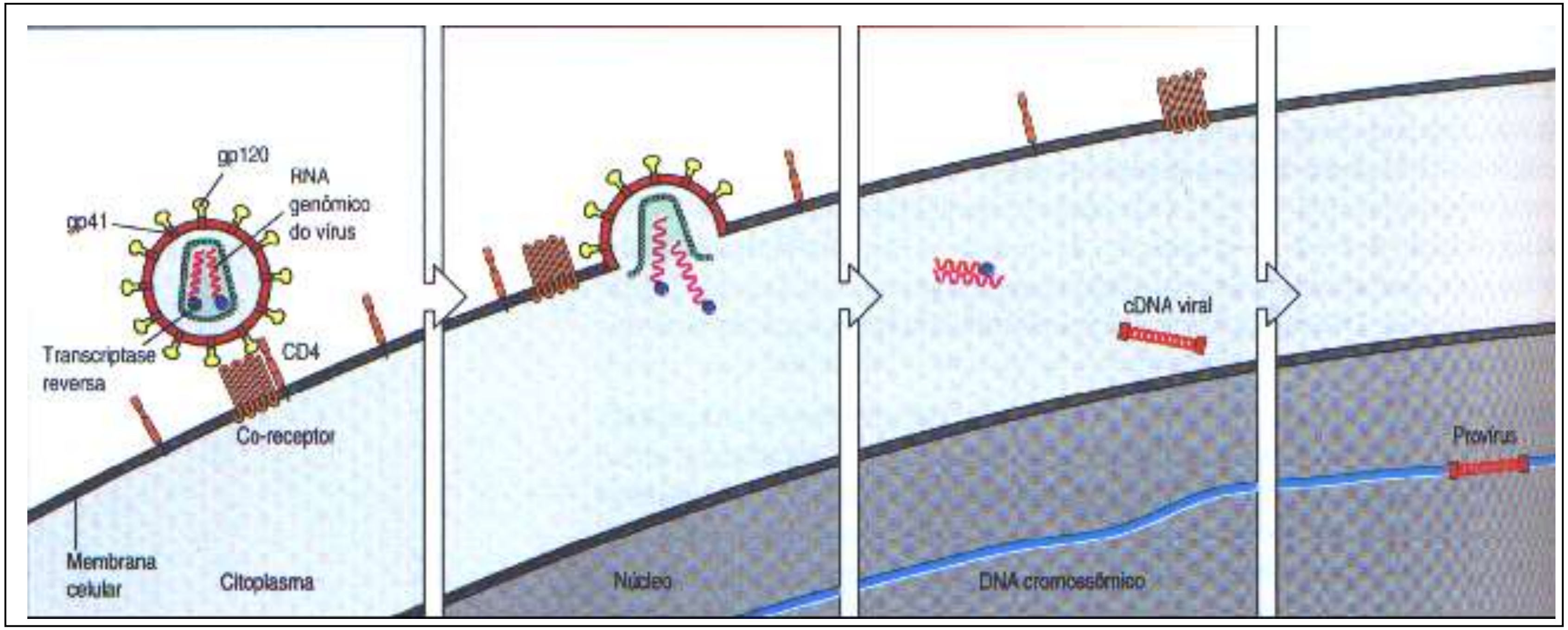


The HIV-1 virion (virus particle) structure includes:

- The genome codified in the RNA.
- Reverse transcriptase (enzime), that allows to transcribe the viris RNA that enters a cell in complementary DNA sequences.
- Protease (enzime), that allows the prodution of new virions.

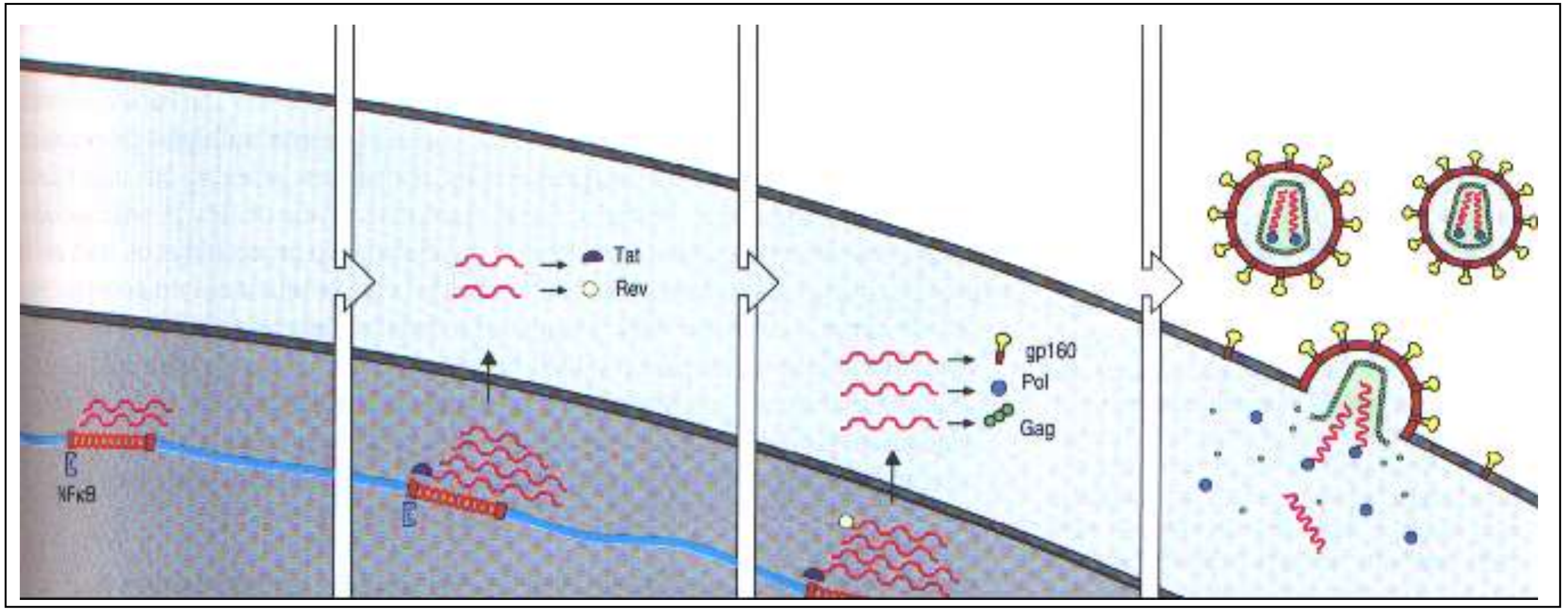
Infection of T-CD4 cells by HIV-1

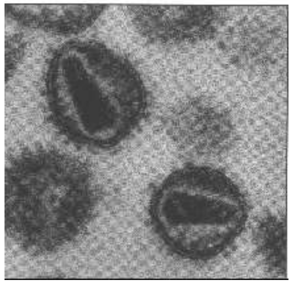
1 2 3 4



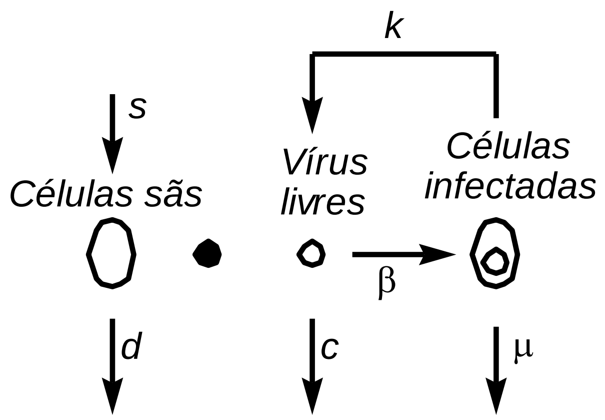
Production of HIV-1 virions by infected T-CD4 cells

5 6 7 8



Dynamics of HIV-1 infection



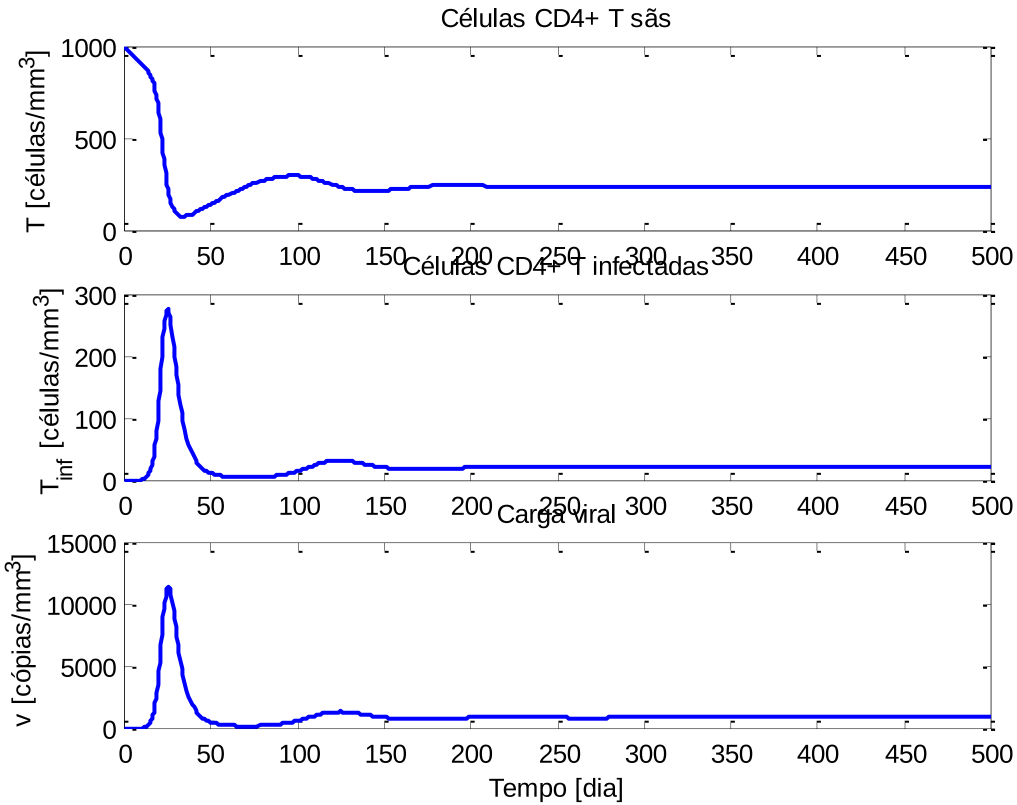
$$\begin{cases} \dfrac{dT}{dt} = s - dT - (1-u_1)\beta T v \\ \dfrac{dT^*}{dt} = (1-u_1)\beta T v - \mu_2 T^* \\ \dfrac{dv}{dt} = (1-u_2)kT^* - \mu_1 v \end{cases}$$

T = Number of healthy T-CD4+ cells per unit volume

T^* = Number of infected T-CD4+ cells per unit volume

v = Number of HIV-1 virion particles per unit volume

Time response of the nonlinear HIV-1 infection model



General form of the non-linear state model

Model of dynamics:

$$\frac{dx_1}{dt} = f_1(x_1, x_2, \ldots, x_n, u_1, \ldots, u_m)$$
$$\frac{dx_2}{dt} = f_2(x_1, x_2, \ldots, x_n, u_1, \ldots, u_m)$$
$$\vdots$$
$$\frac{dx_n}{dt} = f_n(x_1, x_2, \ldots, x_n, u_1, \ldots, u_m)$$

Sensor (observations) Model:

$$y_1 = h_1(x_1, x_2, \ldots, x_n)$$
$$\vdots$$
$$y_n = h_n(x_1, x_2, \ldots, x_n)$$

Nonlinear state model in vector form

Model of dynamics: $$\frac{dx}{dt} = f(x, u)$$

Initial conditions: $$x(0) = x_0$$

Observations model: $$y = h(x)$$

x = State vector (*dim(x)=n*)

u = Manipulated variables vector (*dim(u)=m*)

y = Output (observations) vector (*dim(y)=p*)

Existence and unicity of the solution

A suficient condition for the solution

$$\frac{dx}{dt} = f(x) \qquad x(0) = x_0$$

to exist and be unique is that $\dfrac{\partial f}{\partial x}$ is continuous in a neighborouhood of x_0.

In linear systems, the solution always exists and is unique (why?).

For non-linear systems, it is possible to find examples in which $\dfrac{\partial f}{\partial x}$ is not unique at a point, through which two solutions pass (the theorem is not applicable).

An example of nonunicity of solution

$$\frac{dx}{dt} = x^{1/3} \qquad x(0) = 0$$

This initial value problem has two solutions:

$$x(t) = \left(\frac{2}{3}t\right)^{3/2} \qquad \text{and} \qquad x(t) = 0$$

Observe that

$$\frac{\partial f}{dx} = \frac{\partial}{\partial x}\left(x^{1/3}\right) = \frac{1}{3}x^{-2/3}$$

Is not continuous at $x = 0$ and therefore the theorem may not be applicable.

Hereafter, we assume that the vector field is such that the solution exists and is unique.

Equilibrium states

An equilibrium state is a state such that, if the solution starts on it, it becomes constnt. The corresponding orbit reduces thus to a single point.

Given the equation

$$\frac{dx}{dt} = f(x)$$

the equilibrium states are roots of the algebraic equation

$$f(x) = 0$$

MATLAB functions *fsolve* (Optimization toolbox) or *trim*

The MATLAB *fsolve* function

Aditional info: Help on *Nonlinear Systems of Equations*, *Nonlinear Equations with Analytic Jacobian* e *fsolve*.

Example: Solve the nonlinear algebraic system of equations

$$f_1 = (x_1 - 2)^2 x_2 = 0$$
$$f_2 = (x_2 - 2)^2 x_1 = 0$$

In the work directory create *f.m* with the content:

```
function f=f(x);
f(1)=((x(1)-2)^2)*x(2);
f(2)=((x(2)-3)^2)*x(1);
```

In MATLAB command line give the command *fsolve(@f,[5 6])*

Equilibrium points of linear systems

$$\frac{dx}{dt} = Ax$$

The equilibrium points are given by the roots of the linear, algebraic equation:

$$Ax = 0$$

If the matrix A is non-singular, there is only one equilibrium point $x = 0$.

If the matrix A is singular, there are infinite equilibrium points in a subspace that contains the origin.

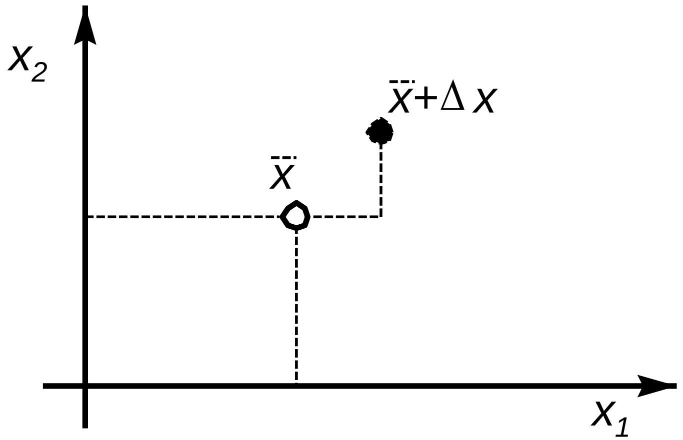
Opposite to non-linear systems, in linear systems there can be no multiple isolated equilibria.

Linearization around an equilibrium point

Let $\bar{x}$ be an equilibrium of

$$\dot{x} = f(x)$$

We want to approximate the dynamics of the increment Δx around the equilibrium point by a linear model.



$$\dot{x} = f(x) \qquad\qquad \frac{d}{dt}(\bar{x} + \Delta x) = f(\bar{x} + \Delta x)$$

Expand in a Taylor series:

$$\frac{d}{dt}\bar{x} + \frac{d}{dt}\Delta x \approx f(\bar{x}) + \frac{\partial f}{dx}\Big|_{x=\bar{x}} \Delta x$$

$$\frac{d}{dt}\bar{x} = 0$$ and $$f(\bar{x}) = 0$$ (why?)

$$\frac{d}{dt}\Delta x \approx \left[\frac{\partial f}{dx}\Big|_{x=\bar{x}}\right]\Delta x$$

Conclusion

The linearization of

$$\dot{x} = f(x)$$

Around the equilibrium state $\bar{x}$ (solution of $f(\bar{x}) = 0$) is given by

$$\frac{d}{dt}\Delta x = A\Delta x \quad \text{with} \quad A = \left[\frac{\partial f}{dx}\Big|_{x=\bar{x}} \right]$$

where Δx is the increment of x around $\bar{x}$.

Linearization around an equilibrium point: Example

$$\frac{dx_1}{dt} = x_1 - x_2$$

$$\frac{dx_2}{dt} = 1 - x_1 x_2$$

The equilibrium points satisfy simultaneously $x_2 = x_1$ and $x_1 x_2 = 1$.

They are (-1, -1) and (1,1).

$$\frac{dx_1}{dt} = x_1 - x_2$$

$$\frac{dx_2}{dt} = 1 - x_1 x_2$$

Jacobian matrix

$$\begin{bmatrix} \frac{\partial f_1}{dx_1} & \frac{\partial f_1}{dx_2} \\ \frac{\partial f_2}{dx_1} & \frac{\partial f_2}{dx_2} \end{bmatrix} = \begin{bmatrix} 1 & -1 \\ -x_2 & -x_1 \end{bmatrix}$$

This matrix must be computed at the equilibrium points.

Equilibrium point (-1,-1):

$$\begin{bmatrix} \frac{\partial f_1}{dx_1} & \frac{\partial f_1}{dx_2} \\ \frac{\partial f_2}{dx_1} & \frac{\partial f_2}{dx_2} \end{bmatrix}_{\substack{x_1=-1 \\ x_2=-1}} = \begin{bmatrix} 1 & -1 \\ -x_2 & -x_1 \end{bmatrix}_{\substack{x_1=-1 \\ x_2=-1}} = \begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix}$$

Equilibrium point (1,1):

$$\begin{bmatrix} \frac{\partial f_1}{dx_1} & \frac{\partial f_1}{dx_2} \\ \frac{\partial f_2}{dx_1} & \frac{\partial f_2}{dx_2} \end{bmatrix}_{\substack{x_1=1 \\ x_2=1}} = \begin{bmatrix} 1 & -1 \\ -x_2 & -x_1 \end{bmatrix}_{\substack{x_1=1 \\ x_2=1}} = \begin{bmatrix} 1 & -1 \\ -1 & -1 \end{bmatrix}$$

These results may be obtained numerically with the MATLAB function *linmod*.

Inferring the stability of an equilibrium point of a non-linear system from the linearization

$$\frac{dx}{dt} = f(x)$$

Assume $x = 0$ is an isolated equilibrium.

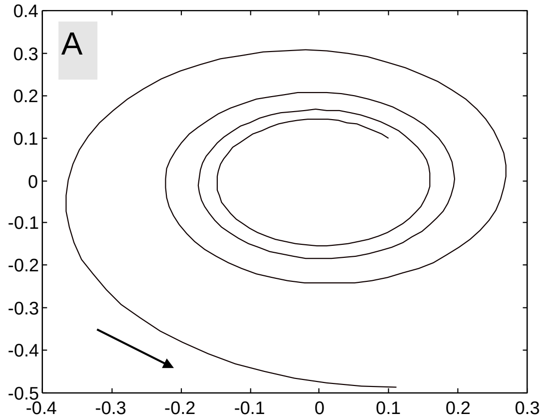
- If all the eigenvalues of the linearized matrix have negative real part, then the equilibrium is asymptotically stable.
- If there is at least one eigenvalue with positive real part, it is unstable.
- If there is at least one eigenvalue over the imaginary axis and all the others have negative real part, nothing can be told about the stability of the equilibrium in the nonlinear system.

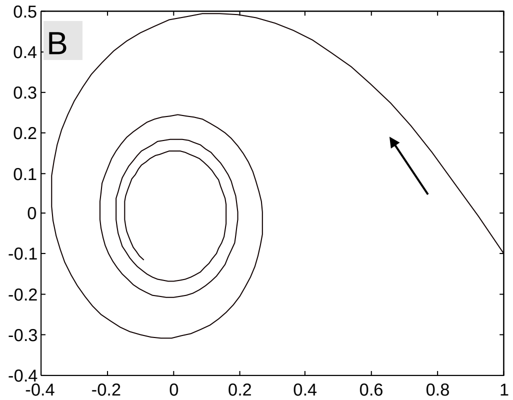
Example where nothing may be inferred

A)
$$\frac{dx_1}{dt} = -x_2 + x_1(x_1^2 + x_2^2)$$
$$\frac{dx_2}{dt} = x_1 + x_2(x_1^2 + x_2^2)$$

B)
$$\frac{dx_1}{dt} = -x_2 - x_1(x_1^2 + x_2^2)$$
$$\frac{dx_2}{dt} = x_1 - x_2(x_1^2 + x_2^2)$$

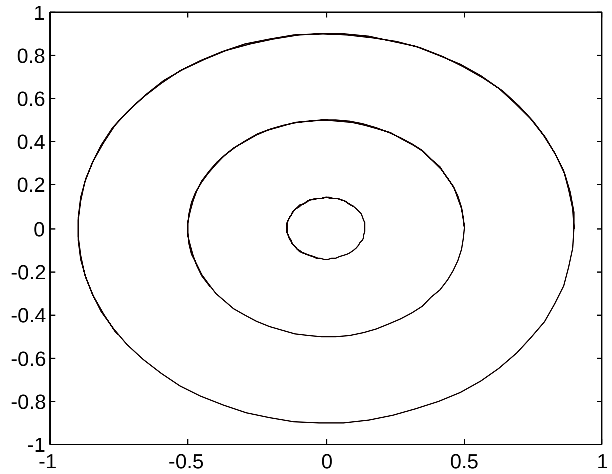
A is unstable and B is stabe





However, *they both have the same linearization* around the origin, given by

$$\frac{dx_1}{dt} = -x_2$$
$$\frac{dx_2}{dt} = x_1$$



The eigenvalues of the matrix of the linearized system are pure imaginary numbers.

Back to the HIV-1 example: Equilibrium points

Equilibrium are obtained equating the derivatives to zero:

$$\begin{cases} \dfrac{dT}{dt} = s - dT - \beta T v = 0 \\ \dfrac{dT^*}{dt} = \beta T v - \mu_2 T^* = 0 \\ \dfrac{dv}{dt} = kT^* - \mu_1 v = 0 \end{cases}$$

Equilibrium states:

No infection (1)

$$\begin{bmatrix} T_1 \\ T_1^* \\ v_1 \end{bmatrix} = \begin{bmatrix} \dfrac{s}{d} \\ 0 \\ 0 \end{bmatrix}$$

Infection (2)

$$\begin{bmatrix} T_2 \\ T_2^* \\ v_2 \end{bmatrix} = \begin{bmatrix} \dfrac{\mu_1\mu_2}{\beta k} \\ \dfrac{s}{\mu_2} - \dfrac{d\mu_1}{\beta k} \\ \dfrac{ks}{\mu_1\mu_2} - \dfrac{d}{\beta} \end{bmatrix}$$

Linearized dynamics

Equilibrium state:

$$\begin{bmatrix} T_0 & T_0^* & v_0 \end{bmatrix}^T$$

Linearized dynamics:

$$A = \begin{bmatrix} -d-\beta v_0 & 0 & -\beta T_0 \\ \beta v_0 & -\mu_2 & \beta T_0 \\ 0 & k & -\mu_1 \end{bmatrix}$$

Numerical example

t		(variable) Day
d		0.02 per day
k		100 per cell
s		10 mm^{-3} per day
β		$2.4 \times 10^{-5} mm^{-3}$ per day
μ_1		2.4 per day
μ_2		0.24 per day

Equilibrium (1) – No infection

Equilibrium point (1):

$$\begin{bmatrix} T_0 & T_0^* & v_0 \end{bmatrix}^T = \begin{bmatrix} 500 & 0 & 0 \end{bmatrix}^T$$

Dynamic matrix of the linearized system:

$$A_1 = \begin{bmatrix} -0.02 & 0 & -0.012 \\ 0 & -0.24 & 0.012 \\ 0 & 100 & -2.4 \end{bmatrix}$$

This matrix has eigenvalues -0.02, 0.2183 and -2.8583. Since there is an eigenvalue with positive real part, this equilibrium is unstable.

Equilíbrio (2)– Infecção (fase assimptomática)

Equilibrium point (2):

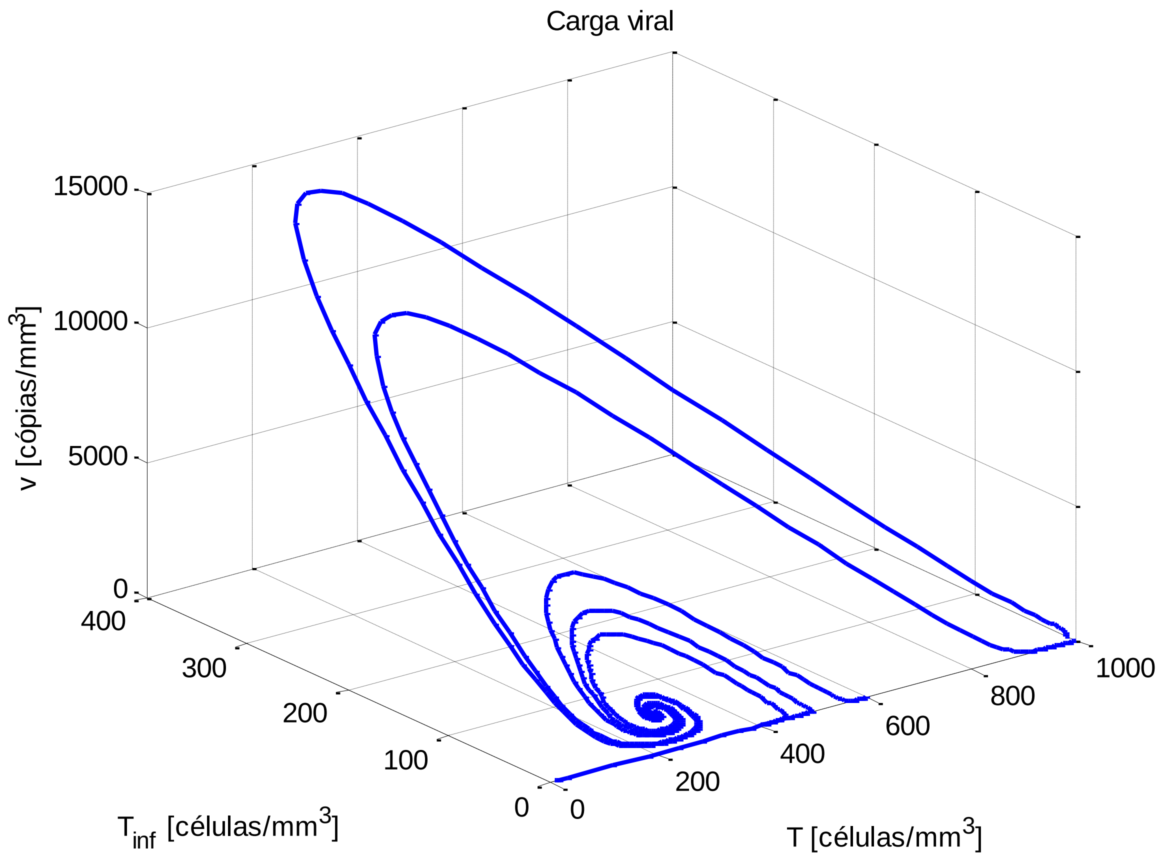
$$\begin{bmatrix} T_0 & T_0^* & v_0 \end{bmatrix}^T = \begin{bmatrix} 240.00 & 21.67 & 902.78 \end{bmatrix}^T$$

Linearized dynamics:

$$A_2 = \begin{bmatrix} -0.0417 & 0 & -0.0058 \\ 0.0217 & -0.24 & 0.0058 \\ 0 & 100 & -2.4 \end{bmatrix}$$

Eigenvalues: $-0.0199 \pm 0.6658\,j$ and -2.6418. Since all the eigenvalues have negative real part, the equilibrium is asymptotically stable.

HIV-1 phase portrait (tridimensional)



The above is not all on the state model (far, far, from it!). It is just a few drops that will help you in the sequel. You need to study linear and nonlinear dynamic optimization. If you didn't follow courses before courses on linear and nonlinear dynamic systems, you may read chapter 4 of the classic book (probably the first course on Optimal Control and still a good one) [AF1966] M. Athans and P. L. Falb, *Optimal Control*, McGraw-Hill, 1966 (There is a Dover reprint) or the beautiful book D. G. Luenberger *Introduction to Dynamic Systems*, Wiley, 1979.

Let's go back to business on dynamic optimization

A basic class of optimal control problems

(Fixed final time, no state constraints)

Let x be the state of a system with manipulated input u, that satisfies

$$\dot{x} = f(x,u) \quad x(0) = x_0 \;\; t \in [0,T] \;\; T \textit{ fixed} \;\; u(t) \in U$$

Find the function u, defined in $[0,T]$ that maximizes

$$J(u) = \Psi(x(T)) + \int_0^T L(x,u)dt$$

L is the lagrangian or running cost

Ψ is the terminal cost penalty

Pontriagyn's Maximum Principle

Along an optimal trajectory of x, u, and λ, the following necessary conditions for the maximization of J are verified:

$$\dot{x} = f(x,u) \qquad x(0) = x_0 \qquad t \in [0, T] \;\; u(t) \in U$$

$$-\dot{\lambda}'(t) = \lambda'(t) f_x\big(x(t),u(t)\big) + L_x\big(x(t),u(t)\big)$$

$$\lambda'(T) = \Psi_x(x)\,\Big|_{x=x(T)}$$

Terminal condition

on the co-state

At each t, the Hamiltonian function H defined by

$$H(\lambda,x,v) = \lambda' f(x,v) + L(x,v)$$

Is maximum for $v = u$ (the optimal control).

Notation:

$$\Psi_x(x)\Big|_{x=x(T)} = \begin{bmatrix} \left.\dfrac{\partial \Psi(x)}{\partial x_1}\right|_{x=x(T)} & \cdots & \left.\dfrac{\partial \Psi(x)}{\partial x_n}\right|_{x=x(T)} \end{bmatrix} \qquad L_x(x,u) = \begin{bmatrix} \dfrac{\partial L}{\partial x_1} & \cdots & \dfrac{\partial L}{\partial x_n} \end{bmatrix}$$

$$f_x = \begin{bmatrix} \dfrac{\partial f_1}{\partial x_1} & \dfrac{\partial f_1}{\partial f_2} & \cdots & \dfrac{\partial f_1}{\partial x_n} \\ \dfrac{\partial f_2}{\partial x_1} & \dfrac{\partial f_2}{\partial x_2} & \cdots & \dfrac{\partial f_2}{\partial x_n} \\ \vdots & \vdots & \vdots & \vdots \\ \dfrac{\partial f_n}{\partial x_1} & \dfrac{\partial f_n}{\partial x_2} & \cdots & \dfrac{\partial f_n}{\partial x_n} \end{bmatrix}$$

The vector λ is called **co-state**, and its equation is the **adjoint equation**.

Other optimal control problems

- More general problems
  - Free terminal time and minimum time problems
  - Final state constraints
  - Other state constraints
- Important special cases
  - Linear dynamics and quadratic constraints
- Bang-bang control and singular arcs

Bibliographic references on OC for the impatient students

- [L1979] Ch. 11, pp. 394 – 435. This a quick and beatiful introduction to the main points of optimal control and dynamic programming, with a justification using calculus of variations – like arguments of the version of the Pontryagin Principle presented above. The whole book is also a very good, easy to read, and sometimes exhilarant, introduction to dynamic systems and control that is strongly suggested to the students with a lack of background on this subjects.
- [R2015] An introduction to the correct formulation of optimal control problems and solving them with Pontryagin Principle. The emphasis is not on mathematical profs, but on developing skills to correctly formulate OC

problems in such a way that they can be solved with numerical packages such as DIDO, for which a free (limited) version is available. The author, I. M. Ross was one of the developers of a class of numerical methods to solve OC problems known as pseudo-spectral methods.

[L1979] D. G. Luenberger. *Introduction to dynamic Systems*. Wiley, 1979.

[R2015] I. M. Ross. *A primer on Pontryagin's Principle . in Optimal Control*. Collegiate Publishers, 2015.

Exercise 2 (Just to warm up)

Design a curve $x(t)$ that starts at $x(0) = 0$, with a maximum slope of 1 and that reaches the maximum height for $t = T$.

The problem may be formulated as an optimal control problem with dynamics

$$\dot{x}(t) = u(t) \qquad x(0) = 0 \qquad U = \{u | u < 1\}$$

and cost functional to be maximized

$$J = x(T)$$

Use Pontryagin's Principle to find the optimal solution.

$$-\dot{\lambda}'(t) = \lambda'(t) f_x\big(x(t),u(t)\big) + L_x\big(x(t),u(t)\big) \qquad \lambda'(T) = \Psi_x(x)\,\big|_{x=x(T)}$$

Since

$$f_x(x,u) = 0 \qquad \text{and} \qquad L(x,u) = 0$$

The adjoint equation is

$$-\dot{\lambda}(t) = 0$$

With terminal condition

$$\lambda(T) = 1 \quad \text{since} \quad \Psi(x(T)) = x(T)$$

Hence

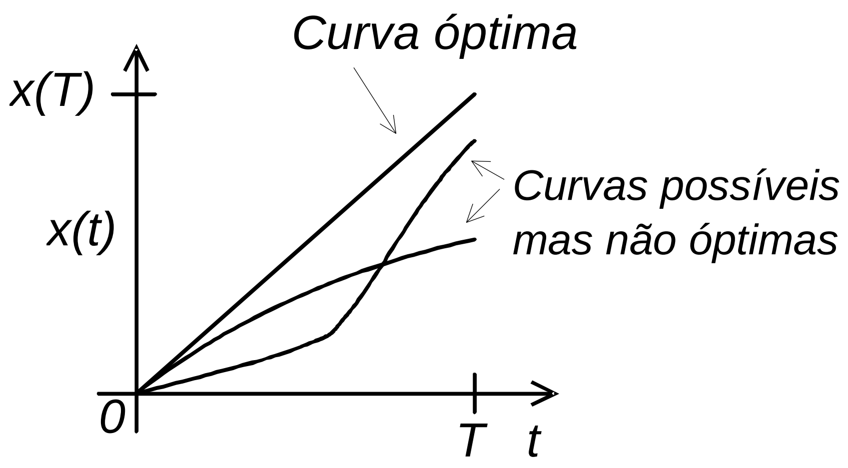
$$\lambda(t) = 1 \qquad 0 \le t \le T$$

The Hamiltonian is

$$H = \lambda' f + L = \lambda u = u$$

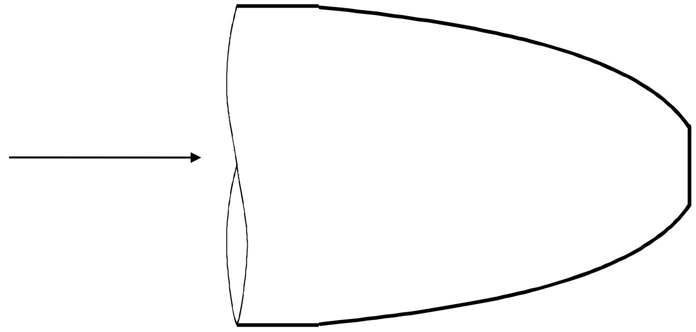
At each *t* the value of *u* that maximes *H* in the set *U* is thus

$$u_{opt}(t) = 1$$

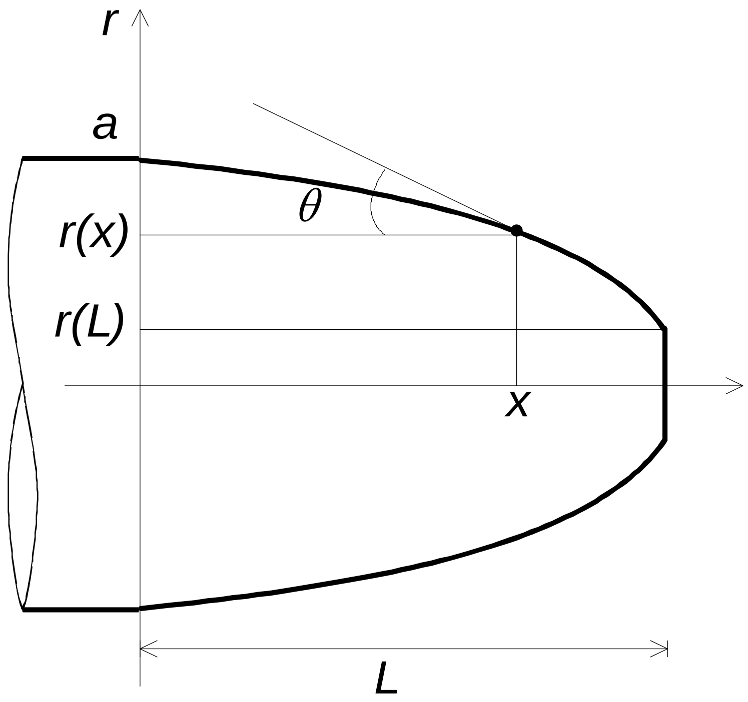


Example: Minimum drag shape of a shell

What is the shape of a shell that leads to a minimum drag?



This problem was solved by Newton in 1686 (10 years before Johann Bernouilli's challenge on the brachistochrone). Newton was aiming an application to ship design but the model he used for the drag force was valid only for very low density atmosphere at a hipersonic velocity.



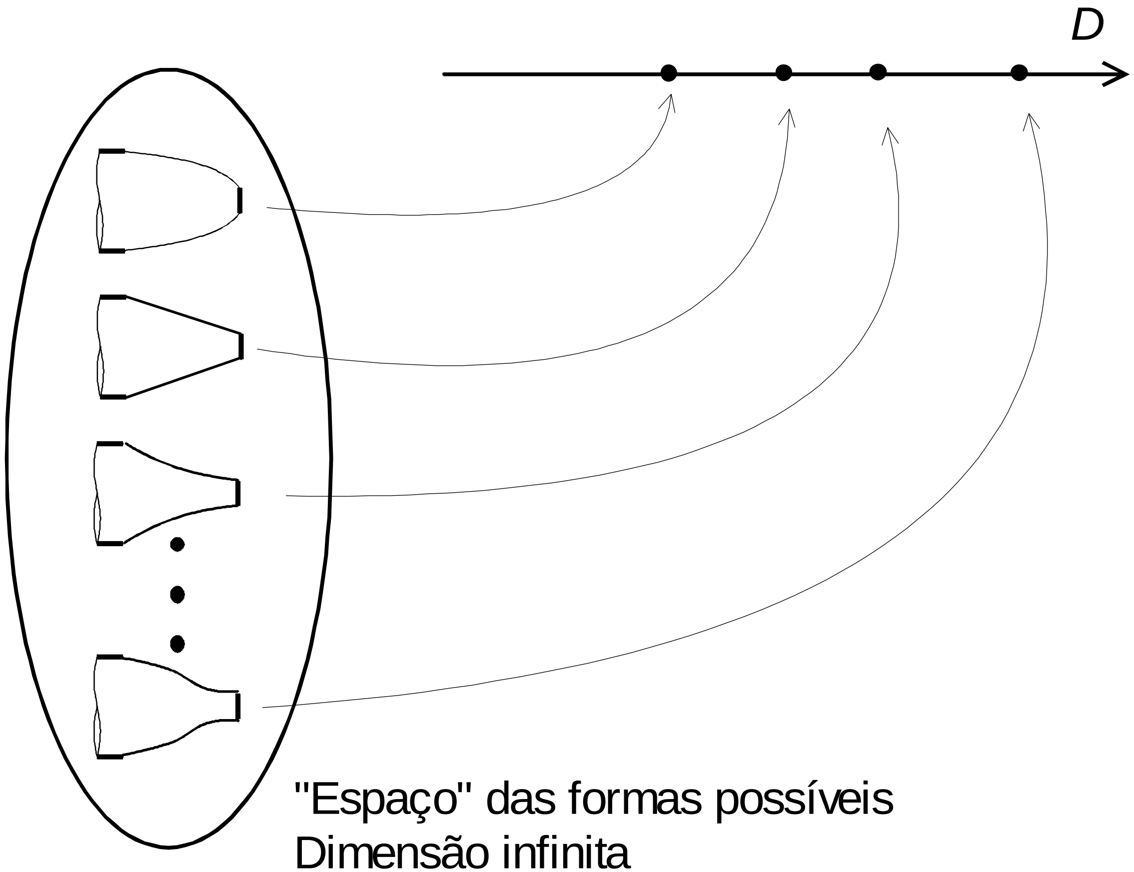
At hypersonic velocities the drag force *D* is approximatelly given by

$$D = -2\pi q \int_{x=0}^{x=L} C_p(\theta) r dr$$

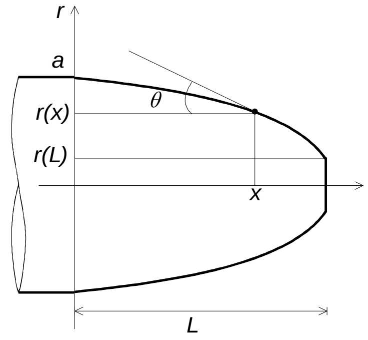
where q is the dynamic pressure assumed to be constant and

$$C_p = \begin{cases} 2\sin^2\theta \; for \;\; \theta \geq 0 \\ 0 \; for \;\; \theta \leq 0 \end{cases}$$

Each shell shape correponds to a drag force.



$$D = -2\pi g \int_{x=0}^{x=L} 2\sin^2\theta\, r\, dr$$



Can be formulated as an Optimal Control problem:

Minimize:

$$\frac{D}{4\pi q} = \frac{1}{2}\left[r(L)\right]^2 + \int_0^L \frac{ru^3}{1+u^2}\,dx$$

Subject to the "dynamics"

$$\frac{dr}{dx} = u$$

The previous problem illustrates two significant issues:

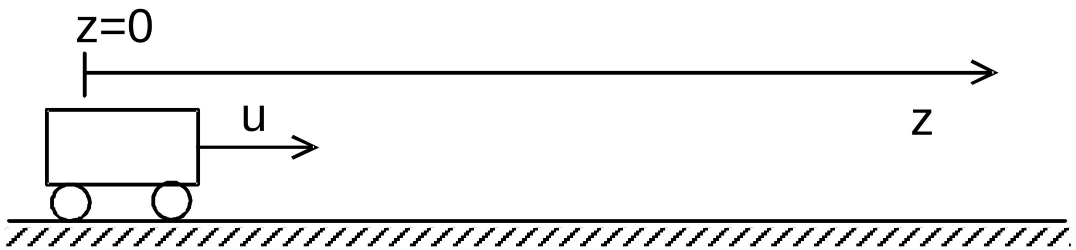
- A shape optimization problem of a planar curve may be transformed into an Optimal Control problem by using a dynamic equation that generates the family of curves considered.
- The problem may be formulated as an Calculus of Variation problem. However, it is readily transformed into an Optimal Control problem by using the dynamic equation

$$\frac{dy}{dt} = u$$

where $y = r$ in the shell problem and u is the control variable.

This technique can be applied to transform a CV problem into an equivalent OC problem.

Exercise 3 – Push cart



Objective: find the function $u(t) \quad 0 \le t \le T$ that maximizes

$$J(u) = x_1(T) - \frac{1}{2}\int_0^T u^2(t)dt \quad , \quad (x_1 := z)$$

sendo a dinâmica do carro dada por (condições iniciais nulas):

$$\frac{d^2 z}{dt^2} = u \quad \text{or} \quad \begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= u \end{aligned} \qquad \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = f(\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}, u)$$

Solution:

$$\begin{bmatrix} f_1(x,u) \\ f_2(x,u) \end{bmatrix} = \begin{bmatrix} x_2 \\ u \end{bmatrix}$$

$$f_x = \begin{bmatrix} \dfrac{\partial f_1}{\partial x_1} & \dfrac{\partial f_1}{\partial x_2} \\ \dfrac{\partial f_2}{\partial x_1} & \dfrac{\partial f_2}{\partial x_2} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}$$

$$J(u) = x_1(T) - \frac{1}{2}\int_0^T u^2(t)dt$$

$\Psi\big(x(T)\big) = x_1(T)$ and hence $\Psi_x\big(x(T)\big) = \begin{bmatrix}1 & 0\end{bmatrix}$

$L(x,u) = -\dfrac{1}{2}u^2(t)$ and hence $L_x(x,u) = \begin{bmatrix}0 & 0\end{bmatrix}$

The adjoint equation is $-\dot{\lambda}' = \lambda' f_x + L_x$ or

$$\begin{bmatrix}-\dot{\lambda}_1 & -\dot{\lambda}_2\end{bmatrix} = \begin{bmatrix}\lambda_1 & \lambda_2\end{bmatrix}\begin{bmatrix}0 & 1\\ 0 & 0\end{bmatrix} \qquad \begin{cases}\dot{\lambda}_1 = 0\\ \dot{\lambda}_2 = -\lambda_1\end{cases} \qquad \begin{bmatrix}\lambda_1(T) & \lambda_2(T)\end{bmatrix} = \begin{bmatrix}1 & 0\end{bmatrix}$$

$$\begin{cases} \dot{\lambda}_1 = 0 \\ \dot{\lambda}_2 = -\lambda_1 \end{cases} \quad \begin{bmatrix} \lambda_1(T) & \lambda_2(T) \end{bmatrix} = \begin{bmatrix} 1 & 0 \end{bmatrix}$$

In this case, the adjoint equation can be solved independently of the state and optimal control. Usually it is not so.

Since

$$\dot{\lambda}_1(t) = 0 \quad \text{we conclude that} \quad \lambda_1(t) = C^{te}$$

From the terminal condition $\lambda_1(T) = 1$ it is concluded that

$$\lambda_1(t) = 1$$

The equation for $\lambda_2(t)$ is

$$\dot{\lambda}_2(t) = -\lambda_1$$

Since $\lambda_1(t) = 1$, this equation becomes

$$\dot{\lambda}_2(t) = -1$$

And hence

$$\lambda_2(t) = C^{te} - t$$

From the terminal condition $\lambda_2(T) = 0$ we get

$$\lambda_2(t) = T - t$$

Hamiltonian:

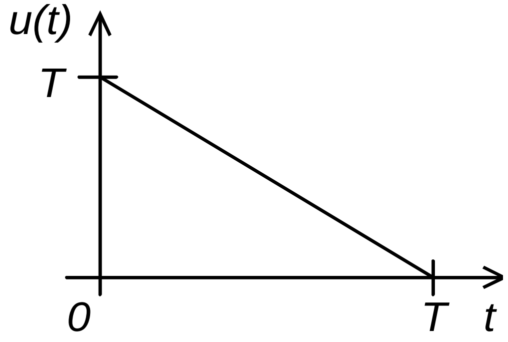
$$H(\lambda, x, u) = \lambda_1 x_2 + \lambda_2 u - \frac{1}{2}u^2$$

In this case there are no control constraints (u may assume values everywhere on $\mathbb{R}$) and the maximum condition for the Hamiltonian is

$$\frac{\partial H}{\partial u} = 0 \quad \text{or} \quad \lambda_2 - u = 0 \quad \text{for each time } t$$

The optimal control is thus

$$u_{opt}(t) = \lambda_2(t) = T - t$$



Exercise 4 – Push cart with minimum fuel

maximize $$J(u) = x_1(T) - \int_0^T u(t)dt$$

s. t. $$\dot{x}_1 = x_2$$ $$\dot{x}_2 = u$$ and $$0 \le u \le \bar{u}$$

Assume $T > 1$.

Solution

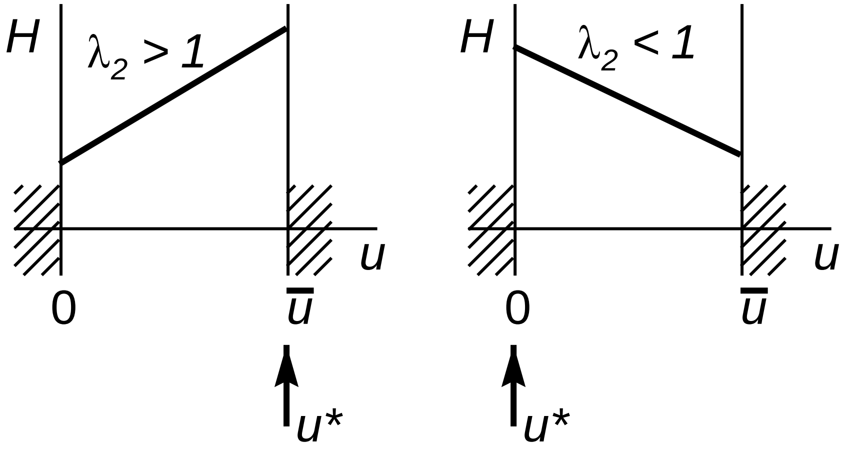
The co-state is as before:

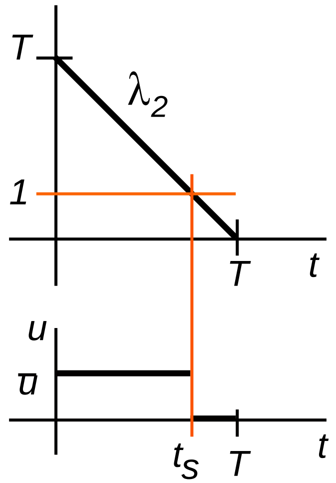
$$\lambda_1(t) = 1,\ \ \lambda_2(t) = T - t$$

The Hamiltonian is now

$$H = [\lambda_1 \quad \lambda_2] \begin{bmatrix} x_2 \\ u \end{bmatrix} - u = \lambda_1 x_2 + (\lambda_2 - 1)u$$

Since the Hamiltonian is linear in u, its maximum is attained at the boundary of the interval of the acceptable values for u.





$$\lambda_2(t_s) - 1 = 0$$

$$T - t_s - 1 = 0$$

$$t_s = T - 1$$