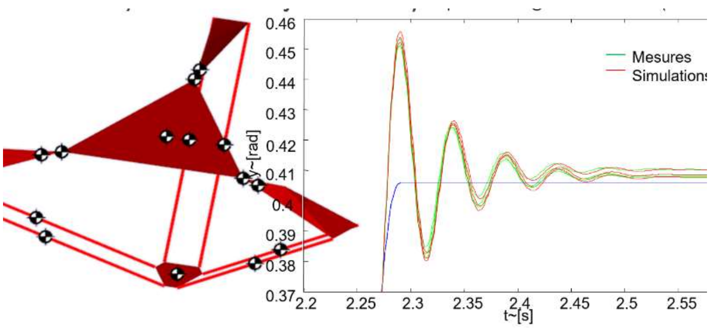
M. Bouri, 2022

PART III

Dynamics

Dynamic modeling of robots for manipulation



ABSTRACT

This chapter presents and develops the tools necessary for modeling the dynamics of a robot. The dynamic model is the basis for analyzing the robot and for synthesizing its controller.

Mohamed Bouri

December 2023- version beta

Contents:

3.1 Introduction.

This chapter presents and develops the required tools to model the dynamics of a robot[1]. The dynamic model is the basis for analyzing the dynamic behavior of the robot and to synthesize its controller. At the end of this chapter, the following questions will have been answered:

- What constitutes or represents the dynamic model of a robot?
- How to proceed to establish such models for a given robot configuration?
- What are its variants and their respective applications (inverse models and simplified models)?
- What are the pros and cons of implementing a dynamic model? Why and when should it be used?

This chapter will discuss the establishment of the dynamic model of robots using two analytic methods, the Lagrange approach applied to serial robots and the Newton-Euler approach that applies to both serial and parallel robots. For pedagogical reasons, these methods will be applied to simple systems.

The following questions will be covered in the control chapter:

- What is the difference between a traditional controller and a controller using a dynamic model?
- What considerations are needed to successfully implement them in robot controls?

3.2 Important definitions and reminders

Geometry deals with the positions (robot configurations) and the relationships that link the different frames of reference

Differential kinematics concerns the velocities and accelerations of the robot's reference frames as well as the transformations that link them together

The dynamics incorporates not only the kinematics but also the masses and the moments of inertias of the structure considered as well as the forces applied (generalized torques).

The generalized coordinates are the variables of displacement of the actuated robot joints. They are called generalized because they may represent translations or rotations depending if the joint is prismatic or angular. These variables are represented by the vector $q(t)$, whose derivatives with respect to time are respectively the generalized velocity vector $\dot{q}(t)$ and the generalized acceleration vector $\ddot{q}(t)$.

Generalized torques are the efforts generated by the robot's actuators so that the end effector performs the desired movement. These efforts can be translational (forces expressed in Newton [N]) or rotational (torques expressed in Newton.meter [N.m]). They are denoted by $\Gamma(t)$.

The robot control is defined as the input required by the actuators and as a correcting command signal which, combined with the desired position, guarantees or ensures that the actuators of the robot are forced to perform a given task in a defined path with some fidelity.

---

[1] We only consider serial and parallel robots for manipulation

3.2.1 Dynamic representations

Modeling consists in obtaining differential equations that represent the evolution over time of an output of a dynamic system Σ.

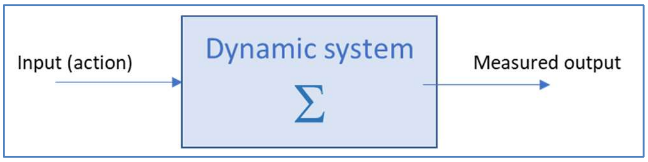
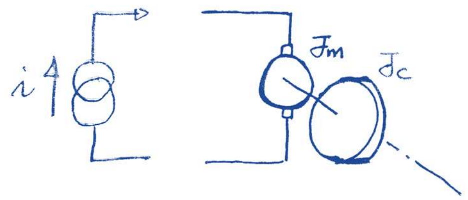
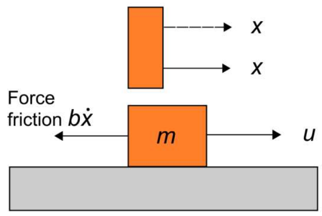


Figure 3.1. Illustration of a dynamic system, with an input variable (action) and a measured output. The input is the control variable or the action that affects the output of this dynamic system.

We recall that in mechanics, the laws of motion can be obtained from the fundamental equations of dynamics. For the translation, we set the balance of the forces which act on the solid for the realization of this translation, and for the rotation we pose the balance of the moments applied to the solid for the realization of this rotation. Let us consider as:

For the translation	$\sum F = ma$	m is the mass in motion and accelerated with the acceleration a	eq. 3.1
For the rotation	$\sum \Gamma = J_R . \gamma$	J_R is the total moment of inertia referred to the axis of rotation accelerated with the acceleration γ	eq. 3.2

3.2.1.1 *Examples that are covered in the slides*

Example 1 - DC motor + load	*Example 2 - Speed control*
	

3.2.1.2 *Basic dynamic systems:*

The basic dynamic behaviors are 1) the integrator and 2) the differentiator. In robot dynamics we will most often find the following dynamic behaviors:

Simple integrator	→ $\int$ →
Double integrator	→ $\int$ → $\int$ →
First-order	→ $\frac{1}{1+\tau \cdot s}$ →
Stable second-order system	→ $\frac{1}{(s/\omega_n)^2 + 2\xi s/\omega_n + 1}$ →

<u>Exercise</u>

Find the transfer function of the dynamic system complying with the following differential equation:

$$I.\ddot{\theta} \; = \Gamma_{in} - \, mgl\, sin\, sin\,(\theta) \; - b.\dot{\theta}^2.sign(\theta) \qquad \text{eq. 3.3}$$

3.2.1.3 Dynamic representations

systems are described by differential equations that we can put in the form of transfer functions if the dynamic is linear, or transfer matrices in the case of systems with several inputs and several outputs. In all cases, the representation in equation of state is possible.

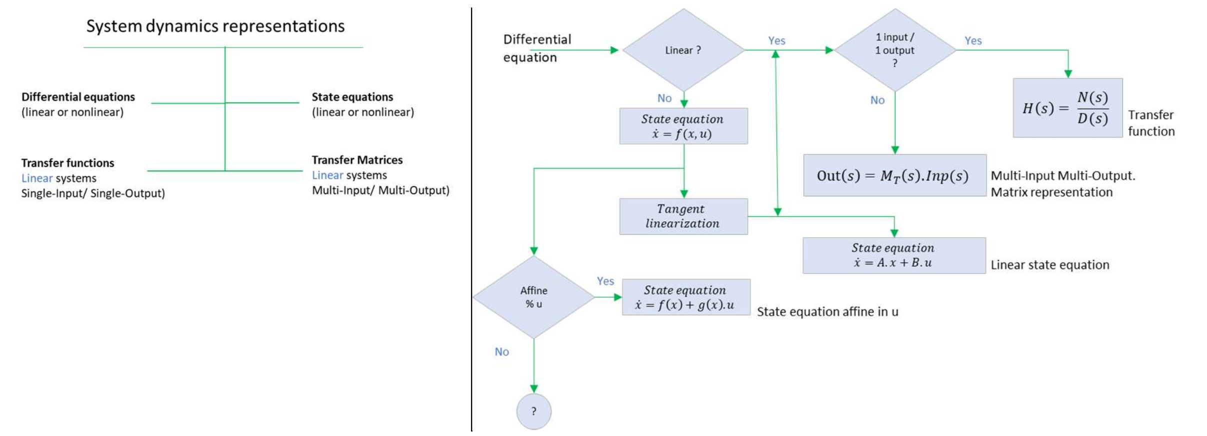


Figure 3.2. (<u>Left</u>) Different system dynamics representations, (<u>Right</u>) tree-methodology function of the dynamics behaviors

<u>Examples</u>

differential equation:	$m\ddot{x} = F_m - b\dot{x} - F_0 - mgl\sin(\beta x)$
transfer function:	$\frac{y}{u} = \frac{1}{1+\tau s}$
equation of state:	$\begin{pmatrix}\dot{x_1}\\ \dot{x_2}\end{pmatrix} = \begin{pmatrix}0 & 1\\ -a & -b\end{pmatrix}\begin{pmatrix}x_1\\ x_2\end{pmatrix} + \begin{pmatrix}0\\ 1\end{pmatrix}u$
nonlinear equation of state	$\begin{pmatrix}\dot{x_1}\\ \dot{x_2}\end{pmatrix} = \begin{pmatrix}x_2\\ -\sin x_1 + \frac{{x_2}^2}{2}\end{pmatrix} + \begin{pmatrix}0\\ 1\end{pmatrix}u$

3.2.2 Example of dynamic model of a rigid body

In order to illustrate the problems posed to establish the model dynamics for a multi-body robot, we will deal with a simple example with only one arm.

This example does not consider any gearbox.

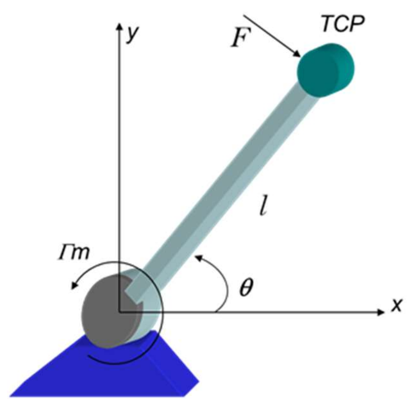


Figure 3.3. Rigid robot axis with an arm of length l and mass m concentrated at the "tool center point": one degree of freedom.

3.2.2.1 Development of the model

For a rigid-body structure, the relations between the position of the terminal effector (Tool Center Point) in the joint space (θ, l) and that in the operational space (x, y) at the TCP are given by:

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} l\cos\theta \\ l\sin\theta \end{bmatrix}$$

And conversely

$$\begin{bmatrix} l \\ \theta \end{bmatrix} = \begin{bmatrix} \sqrt{x^2+y^2} \\ \dfrac{1}{\tan\left(\dfrac{y}{x}\right)} \end{bmatrix}$$

By derivation, we deduce the relations between the speed and the acceleration coordinates along the two axes x and y. First, we obtain for the speed of the center of mass at the TCP as:

$$v = \begin{bmatrix} v_x \\ v_y \end{bmatrix} = \begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} = \begin{bmatrix} -l\cdot\dot{\theta}\sin\theta \\ l\cdot\dot{\theta}\cos\theta \end{bmatrix} = \begin{bmatrix} -l\sin\theta \\ l\cos\theta \end{bmatrix}\begin{bmatrix} \dot{\theta} \end{bmatrix}$$

And the acceleration as:

$$a = \begin{bmatrix} a_x \\ a_y \end{bmatrix} = \begin{bmatrix} \ddot{x} \\ \ddot{y} \end{bmatrix} = \begin{bmatrix} -l\cos\theta \\ -l\sin\theta \end{bmatrix}\dot{\theta}^2 + \begin{bmatrix} -l\sin\theta \\ l\cos\theta \end{bmatrix}\ddot{\theta}$$

Note that the relation between the joint and TCP speeds is linear as we had already noted in the "kinematics" chapter. This is the Jacobian.

The idealized dynamic relation of this robot axis (frictionless, rigid) can be deduced using the Newton-Euler approach (see paragraphs below) or Lagrange. It is given as follows:

$$\Gamma_m - l\cdot F = I\ddot{\theta} + mgl\cos\theta \qquad \text{eq. 3.4}$$

Where:

Γ_m	actuation torque provided by the motor
F	external force acting on the TCP (tool center point)
$I = m\,l^2$	moment of inertia of the arm relative to the axis of the motor
m, l	respectively the mass, concentrated at the TCP, and the length of the rotating arm.
g	the earth's gravity (9.81 N/kg)
θ	Generalized coordinate / simple angle in this case.

This differential equation models the dynamics of our robot axis and can be used to analyze the system as well to develop control.

3.2.2.2 Analysis of the closed-loop dynamic model

If we assume that there are no external forces (F = 0), we may write:

$$\Gamma_m = ml^2\ddot{\theta} + mgl\cos\theta \qquad \text{eq. 3.5}$$

To control the robot arm in position, the simplest solution is a PD controller whose expression for stabilization around the zero setpoint position is given as follows:

$$\Gamma_m = -k_p\theta - k_d\dot{\theta} \qquad \text{eq. 3.6}$$

By introducing the controller into the equation (3.5), we obtain for F=0:

$$I.\ddot{\theta} + k_d\dot{\theta} + k_p\theta = -mglcos(\theta)$$ eq. 3.7

For the controlled robot axis represented by the differential equation (3.7), we find the eigenfrequencies λ based on the characteristic equation:

$$\lambda^2 + \frac{d}{I}\lambda + \frac{p}{I} = 0$$

The eigenvalues of this characteristic equation are given as follows:

$$\lambda_{1/2} = -\frac{d}{2I} \pm \sqrt{\left(\frac{d}{2I}\right)^2 - \left(\frac{p}{I}\right)^2}$$

The system is stable if all the real parts of the natural frequency are negative. This is the case for all $k_d > 0$ (assuming k_p is positive). If one introduces the true parameters of a real system the eigenfrequencies can be quantized and one can adjust the controller parameters (k_p, k_d) as needed.

Synthesis The example of the robot with only one arm shows very simply the dynamic modeling and the corresponding analysis. In this part of the course, a formalism to develop the dynamic model for robots with several degrees of freedom is established. The principle remains the same and the generalization will be discussed.

3.2.3 Forward and Inverse

3.2.3.1 Dynamic Modeling Forward Dynamic Modeling

Question: If we apply the torque profile depicted in Figure 3-3 to the motor in the previous example, what would be the time function describing the evolution of the TCP position?

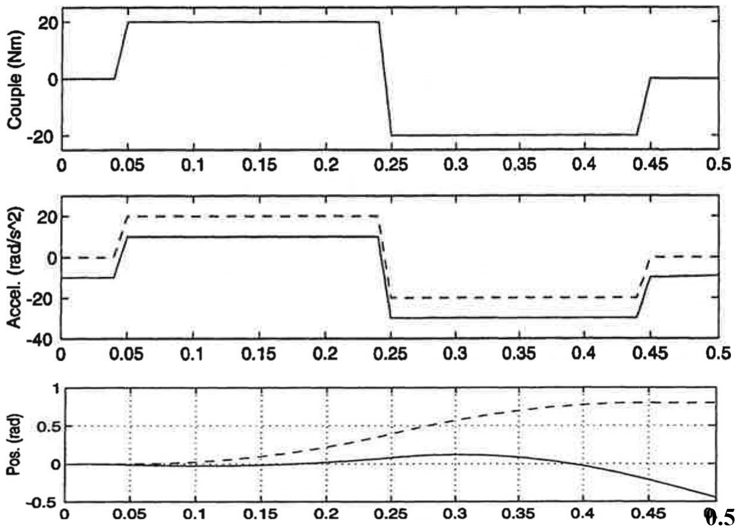


Figure 3.4. Robot output to a predefined torque as a function of time (sec)

This example illustrates the problem set by the modeling of the direct dynamics. It is obvious that the position and its variation are a function of the torque:

$$\theta(t) = f_D(\Gamma_m) \quad \text{eq. 3.8}$$

Where the function f_D is called "Direct Dynamic Model (DDM)". It should be noted that solving this equation requires to integrating the differential equation of the dynamic model.

Modeling direct dynamics: The behavior to a predefined torque amounts to solving a differential equation which corresponds to the expression of the direct dynamic model.
This expression can be presented as an equation of state or as a transfer function, for the particular case of a linear representation. (ref, paragraph Dynamic representations).

3.2.3.2 Inverse dynamic modeling

Question: If we want the position of the end of the robot to move as a function of time according to the graph in Figure 3-4, what will be the time function of the torque to be applied to the motor?

This question illustrates the problem of ***modeling inverse dynamics***. This function, or inverse relationship, relates the evolution of the desired position, speeds and accelerations to the required input torques.

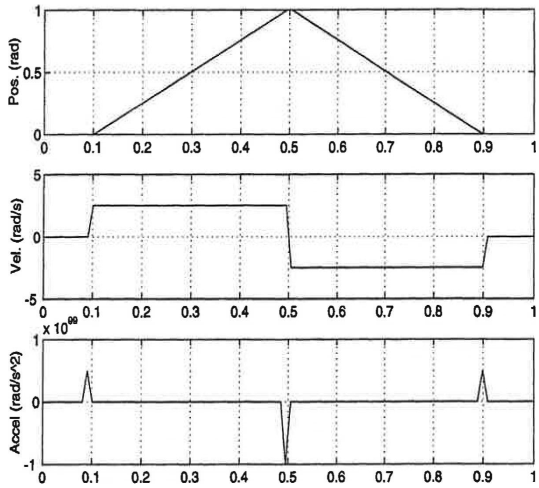


Figure 3-5 Torque required to achieve a predefined motion as a function of time (sec)

$$\Gamma_m(t) = f_I(\theta, \dot{\theta}, \ddot{\theta}) \quad \text{eq. 3.9}$$

Where the function f_I is called the "Inverse Dynamic Model (IDM)".

Inverse dynamics modeling: The algebraic expression of the generalized torque as a function of the values of the generalized position setpoints, generalized speeds and generalized accelerations corresponds to the inverse dynamic model.

3.2.4 Generalization of the MDI and MDD models:

The previous example, as such, is trivial but it clearly shows the problems that we seek to solve in the modeling of the dynamics. In general, these two functions (MDD and MDI) are more complex than for the example. They depend not only on the evolution of the position and the torques but also on their time derivatives.

The attentive reader will have noticed that a relationship exists between these two models.

Indeed, **IDM** = **[DDM]**$^{-1}$ or $f_D = f_I^{-1}$

We can now generalize the definitions given for the dynamic models of robots. The dynamic models shown in Figure 3-5 represent the transformations between generalized torque vectors and generalized displacements. Each element in these vectors corresponds to a degree of freedom.

Figure 3.6. Forward and inverse dynamic models of a robot.

In the majority of cases, the DDM is used in robot simulation (analysis) while IDM is more useful to size the actuation requirements (example Figure 3-6) and positional control of the robot.

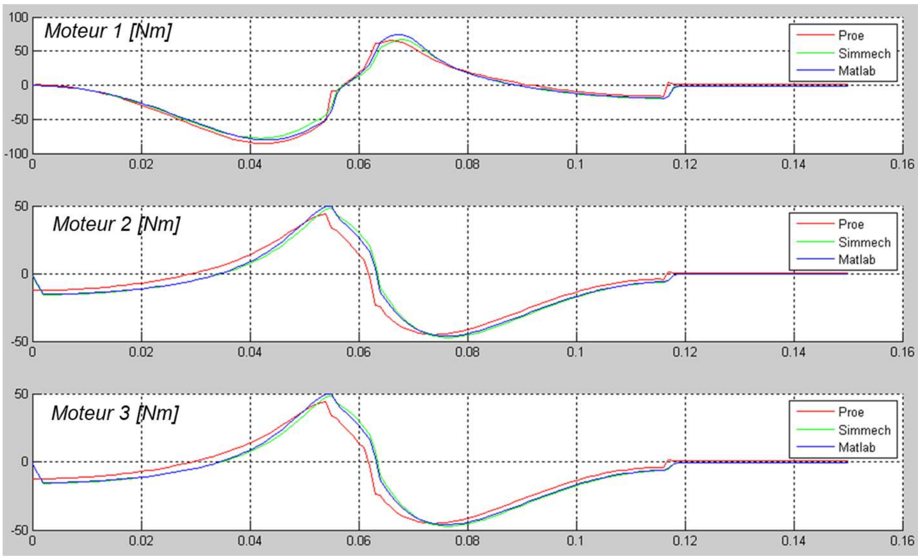


Figure 3.7. *Examples of dynamic torques of a Delta robot following an elliptical pick and place trajectory (acceleration =15Ge and 1kg of load)*

3.3 Generalities about dynamic modeling of a robot

For a given application, the user (the human operator) can explicitly describe the task to be performed. This corresponds to a trajectory to follow generally expressed as a function describing the evolution over time of the robot positions and/or of the speed profile. The torque exerted by the terminal member of the robot can also be used for task description. We therefore understand that it is the inverse dynamic model that first comes to mind when it comes to define the torque control of the robot (eq. 3.9). By using the IDM, the torques a priori necessary to control each actuator, with a view to executing a desired task of the terminal device can be used. These torques or rather their evolutions over time can be evaluated in advance. The purpose of this section is to study the general form of the dynamic equations to highlight the different terms involved. We will discuss their specific properties. These terms, in different forms, are found in all the dynamic models of robots.

The inverse dynamic model of any robot is given by the following general expression

$$\Gamma = B(q).\ddot{q} + G(q) + C(q,\dot{q}) + F(q,\dot{q}) + K(q) \qquad \text{eq.3.10}$$

Γ	the generalized[2] torque vector
$q, \dot{q}, \dot{q}$	the generalized displacement vector and its time derivatives
$B(q)$	the inertia matrix of the robot depending on its position
$G(q)$	the torque vector containing the terms due to gravity
$C(q,\dot{q})$	The torque vector with the terms due to the centrifugal and Coriolis effects
$F(q,\dot{q})$	the torque vector containing the terms due to all the friction effects
$K(q)$	The torque vector that includes the robot stiffness (depending on the position of the robot arms)

This general equation applies to all robots regardless of the type of structure, serial or parallel. Perturbations, by definition, are unknown a priori and therefore cannot be included in this form of equation. However, in practical applications, if the disturbance can be measured before acting on the robot, then their influences can be minimized or even eliminated by an a priori compensation. Certain types of disturbances can be quantified statistically (distribution, variance, and subsequently can therefore be compensated for in a semi-deterministic manner. These disturbances most often appear in the form of random disturbing couples. They are therefore effectively corrected at assistance of torque regulation architectures for which the use of the dynamic model is particularly appropriate

3.3.1 Implementation of dynamic models

An important point to mention concerns the implementation technology and the considerations necessary for the implementation of dynamic models for many methods are exposed in the literature about the digital implementation of controllers. A universal solution does not exist. We will present here some typical examples of the starting points to be considered. The final solution depends on the choices made They vary with each situation to be treated

However, **when carrying out an implementation of the dynamic model**, we have to keep in mind to minimize the power required for the calculation of a dynamic model by keeping the number of operations required minimal, while remaining as faithful as possible to the original model. It is necessary to take into account the hardware available (MIPS, Mflops). In this way, it is possible to calculate the dynamics in real time with short sampling periods. It should be noted that the optimization of the calculations is particularly important if the robot contains a large number of degrees of freedom (>3: complex model), and if high precision and/or high cadences are required. To achieve this objective, several aspects can be considered:

[2] Generalized torques, or generalized forces, can be torques [N.m] or forces [N]- ref. Introduction of this chapter

i) There is a possibiliy to develop equations in explicit forms. This is very useful for simulating and sizing the robot. In the case of a numerical implementation for which the computing power is limited, it will be useful to formulate the dynamic equations in recursive or iterative forms (implicit forms). Indeed, these formulations take advantage of the intrinsically iterative or recursive characteristics of the manipulators such as those imposed by the kinematic constraints on the joints (partially or entirely serial robots). The gain in computation time can be very significant. The results of a study presented in [6] indicate that in general the explicit forms, depending on the number of degrees of freedom (n), lead to solutions with a total number of calculations proportional to ($n + n^2 + n^3 + n^4 + ..$), the efficiently written implicit forms remain proportional to (n). The two forms are strictly equivalent but one is more appropriate for analysis while the other is for numerical evaluations.

ii) A formulation that can be executed efficiently by several processors (parallelism) can significantly increase the execution speed of the algorithms. This can be very interesting in the case of a centralized implementation of the control of several robots, for example in the case of deployment for distributed or collaborative tasks between several robots.

iii) The use of indexed tables associated with interpolation algorithms in which dynamic information evaluated offline is stored. At the base of these applications, we find a modified form of the torque equation. It is known in the literature as the configuration space form. It is given by:

$$\Gamma = B(q)\cdot\ddot{q} + G(q) + C'(q)2[\dot{q}_i\dot{q}_j] + C''(q)\dot{q}_i^2 + F(q)\cdot F''(\dot{q}) \qquad \textit{eq. 3.11}$$

Where:

C', C'', F' and F''	are the decoupled functions of the equation of the original state space
$2[\dot{q}_i\dot{q}_j]\cdot$	is a vector whose components represent the set of products of the velocities of the axes of the robot (linked to the Coriolis effects),
$\dot{q}_i^2$	is a vector whose components represent the set of terms where the speed appears squared (centrifugal effects

All the other symbols represent the same quantities as in the original equation of the state space (eq. 3.1).

The main advantage of this formulation is that all the matrices are functions of an independent variable: the position of the robot. This property makes it possible to calculate these matrices in advance, to store them in memory in the form of tables of indexes, then to use it during the operation in real time. The vectors of speed and acceleration are evaluated in real time directly from the information coming from the sensors. access required is proportional to the speed with which e the robot changes configuration. The number of calculations (interpolations and memory accesses) to be carried out is also proportional to the speed of the movement of the robot. The size of the indexed arrays, which depends on the number of discretized positions and the complexity of the interpolation algorithms, used to quantify the "between-points" spaces is the subject of a compromise intended to balance the memory size well by relative to the number of calculations required.

If these techniques do not provide a solution or are not applicable, then deliberately simplified dynamic models can be used as alternatives. The latter are developed by eliminating terms requiring too much computing power in relation to the contribution they bring to the overall result. Such simplifications are frequent and are found very often in analysis problems such as sinθ=θ and cosθ=1, for θ small. In practice, for robots, these simplifications result in the omission of small masses or inertias, friction effects or effects due to lack of rigidity, etc. However, they introduce modeling errors in the robot control, which must be corrected by a closed-loop position controller. It goes without saying that simplifications vary from case to case.

The various aspects of the final solution also and above all depend on the common sense of engineers. The choice of a good compromise brings immense gains and make the adjustment algorithms all the more suitable for improving the closed-loop performance of position control.

In conclusion, the first phase of the development of a robot controller consists in carrying out tests with classic and simple regulators (PD, PID). Good performances are determined by considering stability, robustness, static and dynamic accuracies, and disturbance rejection for the entire workspace. In cases where the performance of the control encounters limitations, a dynamic control (incorporating a dynamic model) can offer a good solution. Note that the time factor necessary for the development of the dynamic model is decisive. In addition, the dynamic control does not need to be complete: a first implementation could contain only compensation of the inertial effects. This controller would then be continuously and progressively improved in order to meet new requirements (changes to the specifications).

3.3.2 Development of the dynamic model

The rest of this chapter will describe the development of the dynamic model of robots using two different methods, the Lagrange approach and the Newton-Euler approach. It is obvious that **only the methods are different, and not the final result**. The analyzes will be brief and summary, more details are available in the references at the end of this chapter.

The dynamic model can be represented in different coordinates as needed. In the chapter paragraphs, we will focus on the representation in generalized coordinates, that is to say, the coordinates defined by the joints of the robot. The representation in Cartesian coordinates can be deduced using the Jacobian.

To simplify the analysis, the following hypotheses are considered:

- The segments of the structure and the transmissions are all rigid
- The masses and moments of inertia of the mobile segments (links) are constant
- The masses and moments of inertia of the motor rotors are neglected

3.4 Development of the dynamic model by the Lagrange approach

3.4.1 The description of the Lagrangian approach

The formalism of Lagrange, already known from your courses of mechanics or physics, allows to describe the equations of motion of the mechanisms in terms of work of the energy of the system. Lagrange's equation:

$$\Gamma_i = \frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}_i}\right) - \left(\frac{\partial L}{\partial q_i}\right) \quad \text{eq. 3.12}$$

$$L = T - U \quad \text{eq. 3.13}$$

Where:

q_i	is the generalized coordinate corresponding to the joint of index i (eg angle of a robot joint)
Γ_i	is the generalized torque associated with the joint of index i. It corresponds to the sum of the external forces (motor torques, force applied to the terminal device, etc.) and the dissipative forces (friction, internal damping, to which the mechanism is subjected at the joint with index i.
T	Total kinetic energy of the robot
U	Total potential energy of the robot

Using Lagrange's equation, we derive the differential equation which represents the dynamics of a robot (eq. 3.1). Lagrange's method requires to establish the total energy available in the mechanism and the external work supplied to the system. This first requires the following preliminary steps:

1. Selection of appropriate generalized coordinates
2. Identification of generalized torques
3. Establishment of the kinetic and potential energy of all the elements (arms) of the robot

The establishment of the energy equations (point 3) requires the determination of all the positions and the speeds of the individual segments of the robot as well as their masses and moments of inertia. **Finding the positions and velocities** comes back to what was developed in the previous chapters, that is, the knowledge of the transformation matrices and the Jacobian.

Before going into the details on the development of the expressions of the energies, few preliminaries dealing with the Jacobians are recalled. Different types of Jacobian matrices exist. The most important is that which links the joint velocities $\dot{q}_J$ to the velocity vector of the TCP (translational and rotational velocity vector). To develop the kinetic energy, we also need to relate the joint velocities $\dot{q}_J$ to the velocities of the center of mass of the different serial segments of the robot[3]. The expression of the Jacobian J^i of the element (segment) i of the robot is obtained in a similar manner to that of the TCP (ref. Jacobian chapter).

[3] Un robot sériel est constitué de n segments articulés rigides, dits également éléments. Chaque segment rigide est mécaniquement lié à un segment en amont et un autre en aval, sauf pour les cas des premiers et derniers segments.

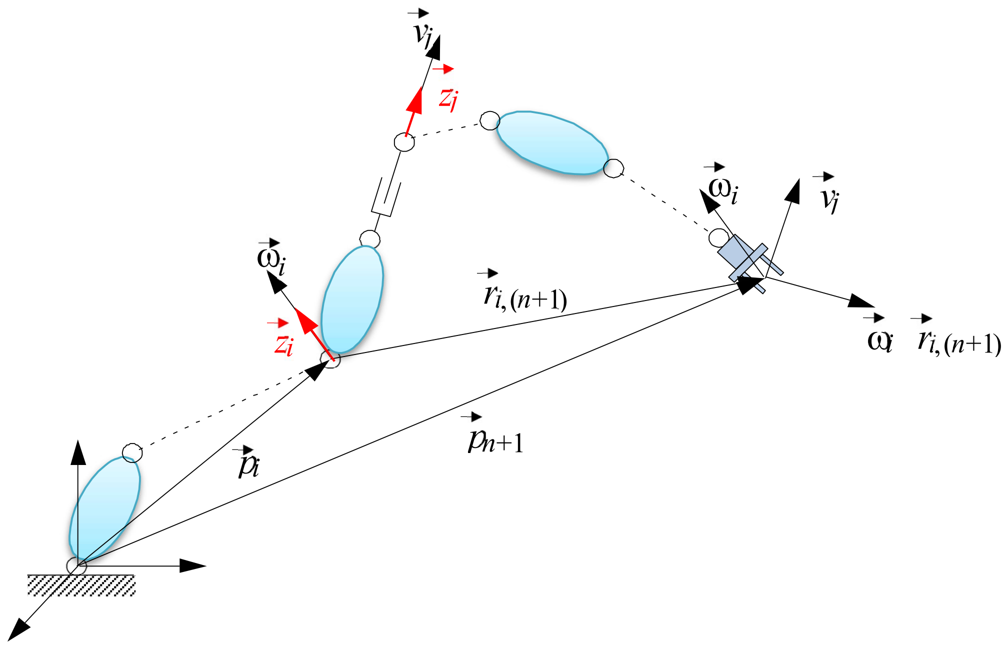


Figure 3.8. Kinematic representation of a serial robot

The Jacobian decomposition for the successive segments of the robot gives us the following:

$$\begin{pmatrix} v_i \\ \omega_i \end{pmatrix} = J^i \dot{q} = \begin{pmatrix} J_p^i \\ J_o^i \end{pmatrix} \dot{q} \qquad \text{eq. 3.14}$$

$$J^i = \begin{bmatrix} J_p^i \\ J_O^i \end{bmatrix} = \begin{bmatrix} J_{p,1}^i & J_{p,2}^i & \dots & J_{p,i}^i & 0 & \dots & 0 \\ J_{O,1}^i & J_{O,2}^i & \dots & J_{O,i}^i & 0 & \dots & 0 \end{bmatrix} \qquad \text{eq. 3.15}$$

Knowing that:

$$v_i = J_p^i \dot{q} \qquad \omega_i = J_o^i \dot{q}$$

The total energy contained in the mechanism is given by the sum of the contributions due to the translational and rotational movements of all the centers of gravity of the different bodies that make up the robot.

The total transitional kinetic energy T_T and the total rotational kinetic energy T_R are expressed as follows:

$$T_T = \frac{1}{2}\sum_{i=1}^{n} m_i v_i^T v_i = \frac{1}{2}\sum_{i=1}^{n} m_i \dot{q}^T J_p^{(i)T} J_p^{(i)} \dot{q} \qquad \text{eq. 3.16}$$

$$T_R = \frac{1}{2}\sum_{i=1}^{n} \omega_i^T \left(R_i I_i R_i^T\right)\omega_i = \frac{1}{2}\sum_{i=1}^{n} \dot{q}^T J_O^{(i)T} \left(R_i I_i R_i^T\right) J_O^{(i)} \dot{q} \qquad \text{eq. 3.17}$$

Where:

I_i	is the moment of inertia of arm i
R_i	is the rotation matrix between the reference frame of inertia (base of the robot) and the reference frame of arm i

The total kinetic energy is given as follows:

$$T_T = T_T + T_R = \frac{1}{2}\sum_{i=1}^{n}\left(m_i \dot{q}^T J_p^{(i)T} J_p^{(i)} q + \dot{q}^T J_O^{(i)T} R_i I_i R_i^T J_O^{(i)} q\right) \quad \text{eq. 3.18}$$

Assuming that the robot arms are perfectly rigid, the potential energy comes from the sole contribution of the forces of gravity.

$$U = \sum_{i=1}^{n} m_i \cdot g^T \cdot p_i \quad \text{eq. 3.19}$$

Where:

m_i	is the mass of the arm i
g	is the vector of gravity represented in the inertial reference frame (base of the robot)
p_i	is the position vector of the center of mass of the segment i with respect to the reference frame of inertia

The inertia matrix $B(q)$ of a robot is a function of the posture of the robot (position of the joints) and is defined by the following expression:

$$T = \frac{1}{2}\dot{q}^T B(q)\dot{q} \quad \text{eq. 3.20}$$

Which we can rewrite in the following scalar form:

$$T = \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} b_{ij}\dot{q}_i\dot{q}_j \quad \text{eq. 3.21}$$

b_{ij} are the elements of the inertia matrix B at row i and column j. The correspondence in equations (3.20) and (3.21) allows to obtain the expression the matrix of Inertia as follows:

$$B(q) = \sum_{i=1}^{n}\left(m_i J_p^{(i)T} J_p^{(i)} + J_O^{(i)T} R_i I_i R_i^T J_O^{(i)}\right) \quad \text{eq. 3.22}$$

The equation (eq. 3.14) will make it possible to deduce the components b_{ij} of the inertia matrix as a function of the kinematics variables (Jacobians and rotation matrices) and the inertias of each segment of the robot. The generalized torque Γ_i , therefore the inverse dynamic model, is then obtained by developing equation (eq. 3.5) thanks to the knowledge of the components of the inertia matrix b_{ij}.

$$\Gamma = \frac{d}{dt}\left(\frac{\partial T}{\partial \dot{q}_i}\right) - \frac{d}{dt}\left(\frac{\partial U}{\partial \dot{q}_i}\right) - \frac{\partial T}{\partial q_i} + \frac{\partial U}{\partial q_i}$$

We obtain:

$$\frac{d}{dt}\left(\frac{\partial T}{\partial \dot{q}_i}\right) = 0$$ (The potential energy depends only on the position)

$$\frac{d}{dt}\left(\frac{\partial T}{\partial \dot{q}_i}\right) = \frac{d}{dt}\left(\sum_{j=1}^{n} b_{ij}\dot{q}_j\right) = \sum_{j=1}^{n} b_{ij}\ddot{q}_j + \sum_{j=1}^{n}\frac{d(b_{ij})}{dt}\dot{q}_j$$

$$= \sum_{j=1}^{n} b_{ij}\ddot{q}_j + \sum_{j=1}^{n}\sum_{k=1}^{n}\frac{\partial(b_{ij})}{\partial q_k}\dot{q}_k\dot{q}_j$$

$$\frac{\partial T}{\partial q_i} = \frac{\partial}{\partial q_i}\left(\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} b_{ij}\dot{q}_i\dot{q}_j\right) = \frac{1}{2}\sum_{i=1}^{n}\sum_{k=1}^{n}\frac{\partial b_{jk}}{\partial q_i}\dot{q}_k\dot{q}_j$$

$$\frac{\partial U}{\partial q_i} = \sum_{j=1}^{n} m_j\left(g\,\frac{\partial p_j}{\partial q_i}\right) = \sum_{j=1}^{n} m_j q J_{p,i}^{j} \qquad \text{(by definition of } J_{p,i}^{j}\text{)}$$

Finally, the expression of the dynamic model of the robot of Figure 3-7 is given by the relation below.

$$\Gamma_i = \underbrace{\sum_{j=1}^{n} b_{ij}\ddot{q}_j}_{a} - \underbrace{\sum_{j=1}^{n}\sum_{k=1}^{n}\left(\frac{\partial b_{ij}}{\partial q_k} - \frac{1}{2}\frac{\partial b_{jk}}{\partial q_i}\right)\dot{q}_k\dot{q}_j}_{b} - \underbrace{\sum_{j=1}^{n} m_j g J_{p,i}^{j}}_{c} \qquad \text{eq. 3.23}$$

3.4.2 Interpretation of the different terms of the dynamic model

The terms (a), (b) and (c) of the expression of the model above (eq. 3.23) contribute to the generation of the dynamic torque at the level of each joint. Recall that Γ_i, represents the vector of generalized forces and the expression remains valid for all robots. This equation can also be represented in a matrix form as:

$$[\Gamma_1, \Gamma_2, \ldots, \Gamma_n]^T = B(q).\ddot{q} + H(q,\dot{q}) \qquad \text{eq. 3.24}$$

The appreciation of these terms (a), (b) and (c) confirms that the inverse dynamic model depends on the generalized joint positions, velocities and accelerations in the following way:

***(a)* On the terms of the accelerations.'**

- The coefficients b_{ii} represent the inertias of joint i if all the other joints are blocked.

The other coefficients (b_{ij}) are associated with the inertial couplings between the axes and express the effects of the acceleration from one joint to another. Note that the couplings between the different joints are often reduced by the choice of robot construction.

(b) Quadratic velocity

- The terms $\left(\frac{\partial b_{ij}}{\partial q_j} - \frac{1}{2}\frac{\partial b_{jj}}{\partial q_i}\right)\dot{q}_j^2$ express the centrifugal forces induced at the joint i by the velocity of joint j. Note that:
  $\left(\frac{\partial b_{ii}}{\partial q_i} - \frac{1}{2}\frac{\partial b_{ii}}{\partial q_i}\right) = 0$ because $\frac{\partial b_{ii}}{\partial q_i} = 0$
- The other terms ($k \neq j$) represent the Coriolis effects induced at joint i by the velocities of joints j and k.

c) Terms due to gravitational acceleration (force)

By introducing non-conservative forces such as viscous friction, dry friction or even a force h at the level of the tool, the expression of the generalized torque is modified as follows:

$$\tau - J^T(q) \cdot h = B(q)\ddot{q} + H(q,\dot{q}) \cdot + F_v\dot{q} + F_s\, sgn(\dot{q}) \qquad \text{eq. 3.25}$$

τ	Is the joint control torque vector
$J^T(q) \cdot h$	Is the projection of the force applied at the tool onto the joints
$F_v\dot{q}$	Is the joint viscous friction force vector
$F_s\, sgn(\dot{q})$	Is the joint dry friction force vector

The relation (eq. 3.25) can also be detailed as below:

$$\tau - J^T(q) \cdot h = B(q)\ddot{q} + C(q,\dot{q}) \cdot \dot{q} + F_v\dot{q} + F_s\, sgn(\dot{q}) - G(q) \qquad \text{eq. 3.26}$$

Where the elements c_{ij} of the matrix C and the gravity vector are defined by:

$$\sum_{j=1}^{n} c_{ij}\dot{q}_j = \sum_{j=1}^{n}\sum_{k=1}^{n}\left(\frac{\partial b_{ij}}{\partial q_k} - \frac{1}{2}\frac{\partial b_{jk}}{\partial q_i}\right)\dot{q}_k\dot{q}_j \qquad \text{eq. 3.27}$$

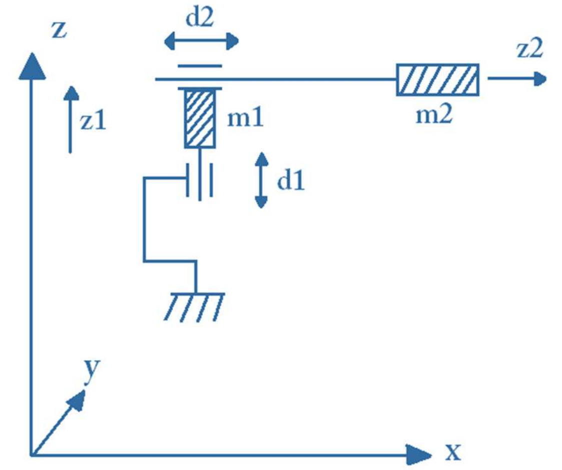
$$g_i(q) = \sum_{j=1}^{n} m_j q J_{p,i}^{j} \qquad \text{eq. 3.28}$$

3.4.3 Presentation of the methodology of the Lagrange's approach for the dynamic modeling of robots

The dynamic modeling approach of robots by the Lagrange method is based on the calculation of the kinetic energies and potentials and on the dissipation of these different energies for the realization of the movement. However, by observing in detail the result of the relations (eq. 3. 18) to (eq. 3. 21), it emerges that there is no need to calculate these energies to deduce the expression dynamic torque. Lagrange's method for obtaining the dynamic model of serial robots is a systematic method that must be implemented according to the following steps:

1. Describe the robot
   a. Identify the tool and the joint variables.
   b. Identify the segments of the robot.
2. Define the base frame of the robot.
3. Define the positive and negative directions of each joint.
4. Write the local Jacobians with respect to each robot segment.
5. Deduce the Matrix of inertia
   a. Thanks to the inertial elements of each robot segment (mass and inertia tensor)
   b. Thanks to the Jacobians reported to each robot segment (pnt 4).
6. Deduce the different components of the dynamic equation (Coriolis, Centrifugal and Gravity).

3.4.4 Exercise, dynamic modeling of a Cartesian robot with 2 axes using the Lagrange method



- Deduce the dynamic model of the Cartesian robot with 2 axes from the figure opposite. **m1** and **m2** are the moving masses of the two horizontal **x** and vertical **z**.
- Deduce the block diagram representation of the dynamics of this robot.
- Write the robot's equations of state.
  Solution, see exercises booklet

3.5 Development of the dynamic model by the Newton-Euler approach

The Newton-Euler formalism is based on the general theorem of mechanics which applies the balance of efforts or inventory of forces acting at the center of mass of each segment of the manipulator.

Newton's equation $f = m\dot{v}_c$ eq. 3.29

Euler's equation $\mu = I_C\dot{\omega} + \omega \times I_C\omega$ *eq. 3.30*

Where:

μ, f	are the vectors of the forces (or torques) acting on the centers of mass of the considered body.
I_c	is the tensor (moment) of inertia of a body
$-_c$	is the index indicating a measured quantity relative to a reference frame placed at the center of mass of a chosen body (segment of the robot)
X	represents the operation of a vector product

The analytical power of this method lies in the development of these two equations, which leads to the following two iterative relations (3.31) and (3.32):

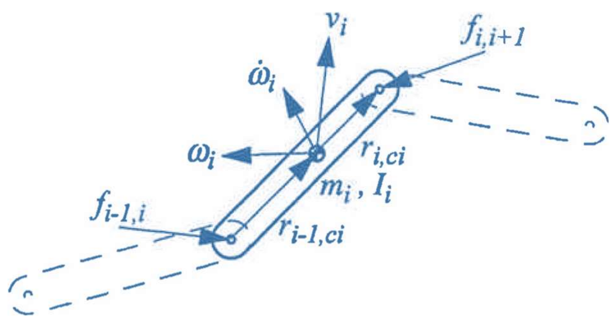


Figure 3.9. forces acting at the center of mass of a body i of the robot

$$f_{i-1,i} - f_{i,i+1} + m_i g - m_i\dot{v}_{ci} = 0 \quad i \in [0, n]$$ eq. 3.31

Where:

i	is the index of the conisdered body (segment)
$f_{j,k}$	is the force of interaction between two bodies of respective indices j and k. interaction with the upstream and downstream bodies
m_i	represents the mass of the considered segment of the robot
g	is the gravitational acceleration vector
v	is the linear velocity vector of the center of mass of the segment i

The Euler equation applied to the segment i, for its displacement in rotation, is the following:

$$\mu_{i-1,i} - \mu_{i,i+1} + (r_{i,ci} \times f_{i,i+1}) - (r_{i-1,ci} \times f_{i-1,i}) - I_{ci}\dot{\omega}_i - (\omega_i \times I_{ci}\omega_i) = 0 \quad i \in [0,n] \qquad \text{eq. 3.32}$$

Where:

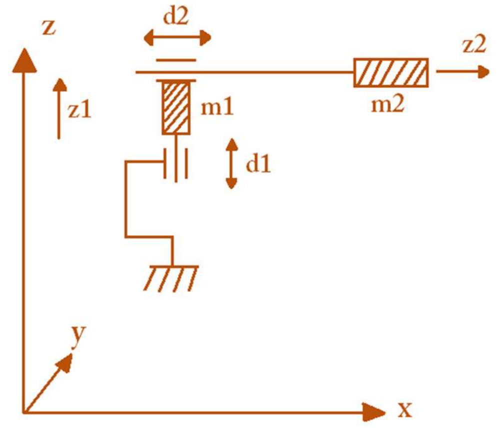
μ_i	is the interaction torque between two upstream and downstream bodies $(i, i+1)$ $(i-1, i)$
$r_{j,ck}$	is the position vector allowing the location of the center of mass of the k^{th}
	body with respect to the origin of the j^{th} reference
I_i	is the matrix of inertia with respect to the center of gravity

3.5.1 Lagrange approach for the dynamic modeling of robots: Presentation of the method

The modeling approach by the Newton Euler's method requires splitting the robot into distinct articulated segments. Newton's equation (eq. 3.22) and Euler's equation (eq. 3.23) are applied to each body. It is obvious that Euler's equation does not apply to bodies in pure translation and that Newton's equation does not apply to bodies in pure rotation around a fixed pivot. The application of this approach can thus be summarized by the following procedure:

1. Describe the robot -
    a. Identify the tool point (TCP) and the joint variables.
    b. Identify the segments of the robot.
2. Define the base frame of the robot.
3. Define the positive and negative directions of each joint.
4. Write Newton's and Euler's equations for each segment.
5. Solve the system of equations formed by this set of equations, and deduce the dynamic model that provides the expression of the joint torques for each segment.

3.5.1 Exercise, dynamic modeling of a 2-axis Cartesian robot using the Newton-Euler method



- Deduce the dynamic model of the 2-axis Cartesian robot from the figure opposite. **m1** and **m2** are the moving masses of the two horizontal **x** and vertical **z**.

<u>**Solution**</u>, see exercise booklet

3.6 Discussion on the two approaches

We have highlighted the analytical power and the equivalence of the two approaches, of Lagrange and Newton-Euler, to establish the dynamic model of a robot, Both techniques consider of punctual point of view the bodies. The Newton-Euler method takes into account the reaction forces between neighboring bodies which is useful in certain cases; while Lagrange's method does not take these forces directly into account in its calculations but only considers the input work, the stored energy and judiciously chosen generalized coordinates. The choice of one or the other of these two methods depends on factors that go beyond the scope of this text. They will however, provide the same end result.

Many other methods can still be found in robotics books. All these techniques lead to formulations which vary in their form but not in their content, this of course if identical starting hypotheses are used. These mathematical models, whose complexities depend on the total number of degrees of freedom of the system, can be directly implemented in micro-processors, as well as computers or microprocessors.

3.7 Other Aspects of the Robot Dynamic Model

Up to a certain point, quasi-linearized approaches to robot control are justified, but their application limits must be well assessed and understood. To perform a movement of the end effector, it is necessary for the actuators to supply an appropriate movement, thanks to an appropriate actuation torque. The mechanical structure is also subject to parasitic torques. The complete dynamic model includes the coupling of inertias, the effect of gravity (weight of the motors, arms and joints), the Coriolis and centrifugal effects, the different types of friction (in the motors and in the joints), the elasticity of structure, etc. <u>All these effects</u>, the list of which is far from complete, are important to consider in the majority of situations but very difficult to completely compensate. **They are all neglected in the simplified model presented above**. The explanation is to be found on the one hand in the complexity of implementing the complete dynamic model considering the hardware used in a traditional control and on the other hand in the fact that certain simplifying hypotheses are easily justifiable. For example, the effects of centrifugal and Coriolis forces are often so weak at operating speeds that they have little influence on the actuated torque Γ. A really complete and detailed model, should take into account not only the aforementioned effects but also those due to the intrinsic properties of the motors such as frictions, limits in the characteristics, cyclic variations of torque or output speed due to errors or manufacturing limits.

It goes without saying that an absolutely complete robot model, intended for use either in simulation or to be incorporated into a controller, should take all these effects into account. The decision to include or not an element depends on the capacity of the control designer to evaluate in a precise manner the absolute and relative magnitudes of these contributions (parasitic components and useful component). The determination of the respective importance of these contributions must be carried out throughout the working phase of the robot, in other words it is a question of evaluating these contributions for all the combinations of speeds, accelerations and walking reversals in the whole region which composes the working volume of the robot. Let us return once again to the linearized dynamic model whose classical equation was shown before. The main motivation for considering only the inertial effects and/or their rates of change, is that in the majority of cases the corresponding torque (useful torque) is the one that predominates. Such a model can be considered as a partially complete dynamic model. Depending on the criteria of the application considered and the robot used (its mechanical design) this method may be sufficient. A general rule is that in order to quantify the quality of a model, it must be possible to attribute either by means of theory, or preferably by practical tests, orders of magnitude to each of the dynamic effects. The amplitude of these quantities, as well as their rates of variation, will determine where the modeling effort should be concentrated.

It goes without saying that to compensate for deviations in performance, due to an incomplete model, during on-line operation we will need to include an accompanying regulator to make the corrections. The Robot Control chapter will return to this subject. It is very important to remember that a regulator is blind because it has no notion of the dynamics of the system. Therefore, it still imposes limitations on the performance possible with the robot.

The more dynamic information incorporated into the controller, the better the robot will react.

3.8 Discussion and conclusions:

A very important phase in the design and development of a robot controller is the analysis, preferably carried out by simulation. This is most easily implemented digitally using a computer with adequate computing power.

It should be remembered that the direct dynamic model is supposed to be a bijection between a numerical representation and the various physical phenomena that take place in the mobile structure of the robot. The flexibility of this numerical representation is such that it allows engineers, using simulations, to have a very realistic understanding (idea) of the behavior of the robot. This behavior can be studied for the entire range of desired speed or torque setpoints.

The objective of the dynamic control is to impose on the robot a planned operation. The choice of the appropriate dynamic algorithm for the robot varies from case to case, because the factors to be considered, which we have mentioned in this brief text, vary from robot to robot, even for two robots of the same model. The predominance of one of the dynamic effects, eg inertial, frictional, gravitational or other, immediately indicates where the emphasis should be placed when designing the control.

The great diffusion and the low costs of microprocessors, favored their introduction in the material used by the techniques of the automatic, which allowed an almost unlimited sophistication of the commands of the robots.

Basically, the controllers react only and only to the state of the deviation between the setpoint and the desired point, associated with the task to be performed. These controllers whose underlying theories are well developed are seen in control theory courses. Several works have made it possible to guarantee their performance by the existence of wide stability margins during their application in robot controls. A main distinction between traditional approaches and those incorporating dynamic aspects lies in the generation and use of information inside the controller. With dynamic controls information is generated a priori to be used to correct expected errors before they occur. Regulators in the pure sense correct errors after they have occurred. This means that in highly dynamic robots (the torques of the motors that we want to achieve as well as their variations are very important), the dynamic commands will always be potentially faster compared to conventional regulators. One such example is illustrated by the direct-drive DELTA robot. At each sampling instant a controller can only see while a dynamic control can predict. The use of inverse dynamic models does not exclude their associations with classical regulators (for example PID), in order to make the most of these two domains, as currently used in the control of the DELTA robot (see [6] and [7]).

The search for an adequate control for a given application requires a lot of common sense and a sufficient knowledge of the orders of magnitude not only physical, but also of mechanical construction, and of course economic. All sources of information are helpful; factors as diverse as the ratios of inertia, the choice of materials, the desired phase of the application (speed limits, more critical regions at the desired spots,...), or any other constraint, are useful for decisions to take. All these together with the simplifying assumptions constitute aspects of considerable importance in the design of efficient robot control!

3.9 References

[1] Harushiko A., Slotine JJ, Robot Analysis and Control, John Wiley and sons - 1985

[2] Sciavicco, L, Siciliano, B., Modeling and Control ofRobot Manipulators, The McGraw-Hill Companies, Inc., 1996

[3] Craig, JJ: Introduction to Robotics: Mechanics and Control, (2nd ed.), Addison-Wesley, 1989

[4] Dombre E., Khalil W., Modélisation et commande des robots, Editions Hermès, 1988.

[5] Liégois A. et Aldon MJ: Génération et Programmation des équations de Lagrange des robots et manipulateurs, Rocquencourt - 1980

[6] Codourey A, Contribution à la commance des robots rapides etprecis, (Application au robot DELTA), these No. 922, 1 991 (EPFL)

[7] Clavel, R., Conception d'un robot parallèle rapide à 4 degrés de liberté, thèse No. 925, 1 991 (EPFL)

[8] Shaninpoor M., A robot Engineering Textbook, Harper and Row - 1987