




Open Archive Toulouse Archive Ouverte

OATAO is an open access repository that collects the work of Toulouse researchers and makes it freely available over the web where possible

This is an author's version published in: <http://oatao.univ-toulouse.fr/23634>

To cite this version:

Noureddine, Farid  *Fault tolerance of robot manipulators*. (2004)
In: AVCS'04, International Conference on Advances in Vehicle Control and Safety, I3M, IFAC, 28 October 2004 - 30 October 2004 (Genova, Italy).

Any correspondence concerning this service should be sent to the repository administrator: tech-oatao@listes-diff.inp-toulouse.fr

FAULT TOLERANCE OF ROBOT MANIPULATORS

F. Noureddine

*Ecole Nationale d'Ingénieurs de Tarbes
BP 1629, 65016 Cedex, France
Email : farid.noureddine@enit.fr*

Abstract: This work is about the study of a fault tolerant rigid robot manipulator having five degrees of freedom. All the stages in the design of a fault tolerant system are described here. In order to simplify the fault detection process and isolation module only faults affecting joint 3 are considered. Fault accommodation relative to a locked joint 3 is established thanks to the kinematic redundancy of this robot.

Keywords: Fault detection in robots, fault detection and isolation, fault accommodation, fault tolerance, robotics

1. INTRODUCTION

Fault tolerance is one of the means available to enhance dependability in systems. Any technological system can prove to have malfunctions despite a decent design. In some cases, these malfunctions can have disastrous consequences. Fault tolerance is mainly aimed at foreseeing the tasks to be performed in order to ensure the continuity of the system in a normal working mode despite the faults and even, to some extent, prevent dangerous situations to be generated by the blocking of joint 3 or any other part of the robot. In order to be real, fault tolerance requires the design of two main stages. Fault detection and isolation, FDI, which will allow the generation of indicators showing the occurrence of faults is called residuals. The second stage should allow the accommodation of the faults that have been detected and isolated and thus permit the continuity of a given task. The preliminary phase in the design of algorithms is the phase consisting in analysing the different types of faults which can affect the whole system and yet allow the creation of a model with faults.

As most of technological systems, robots are subject to faults. According to the context in which the robot is used and the environment of the robot, detected faults could have repercussions that are more or less serious. A detailed analysis of the robot will permit one to anticipate the faults and failures which may occur. See (Visinsky *et al.*, 1995) for a study of fault tolerance in robotics, refer to (Filaretov *et al.*, 2002) for more recent works focussing on FDI in robotics.

Once the fault has been detected and isolated, propositions of fault accommodation can then be defined. The specificity of fault tolerant systems is to design in line and even in real time accommodation procedures. The specificity of robot manipulators is such that their task generally includes end-effector motion in the execution of a precise task. The task is generally described in a task space (cartesian space) and the consequences of the fault must be analysed in this same space. So a fault occurring on a joint can cause it to be blocked and the loss of a freedom degree will significantly reduce the workspace of the robot. Nevertheless it is still possible to consider the robot as being functioning correctly inspite of its performances

being degraded. The different design and implementation stages of a fault tolerant robot will be presented in this paper. The first part is an account of the dynamic modelling of the robot with fault analysis and explains the way it can produce a model with faults. The second part enumerates some points related to the command control as well as to algorithms of fault detection; The last paragraph poposes a solution to fault accommodation in order to reduce the repercussions of a fault.

2. DYNAMIC EQUATIONS OF THE ROBOT AND FAULT MODELLING

2.1 Modelling-Actuation scheme

The considered robot has five degrees of freedom and all joints are rotational (articulated arms). Its dynamic model is given in (Canudas *et al.*, 1997), where $\Gamma \in \mathbb{R}^5$ is the generalized joint torque vector and is expressed by $\Gamma = H(q).\ddot{q} + C(q, \dot{q}).\dot{q} + g(q)$

where

$H(q) \in \mathbb{R}^{5*5}$ is the inertia matrix,

$\ddot{q}, \dot{q}, q \in \mathbb{R}^5$ is respectively the joint acceleration, joint velocity and joint position,

$C(q, \dot{q}).\dot{q} \in \mathbb{R}^5$ is the Coriolis and Centripetal forces,

$g(q) \in \mathbb{R}^5$ is the vector of gravity forces.

The symbolic calculations relative to the different terms of Γ have been implemented with the software package Symoro+, (Khalil and Creusot, 1997).

The actuator of each joint is a Dc motor (permanent magnet motor). These motors are equipped with gears and brakes for joints 2 and 3. Once the dynamic model is obtained, the fundamental law of dynamics, taken into account the viscous friction, the gear ratio and the motor inertia, leads to obtain $\Gamma_m = I_a.\ddot{q}_m + F_v.\dot{q}_m + N^{-1}\Gamma$

$\Gamma_m \in \mathbb{R}^5$ is the vector of the motor torque,

$I_a = \text{diag}\{I_{a5}\}$ is the diagonal matrix of the actuators inertia,

$\ddot{q}_m, \dot{q}_m \in \mathbb{R}^5$ is respectively the actuator acceleration, and actuator velocity,

$F_v = \text{diag}\{F_{v5}\}$ is the diagonal matrix of the parameters of the viscous friction,

$N = \text{diag}\{N_5\}$ is the diagonal matrix of the parameters of the gear ratio.

Thanks to the gear ratio it can be deduced for each joint

$\dot{q}_m = N.\dot{q}$. Moreover the motor torque is calculated according to

$\Gamma_m = K_{c_i}.K.v$, with

$K_c = \text{diag}\{K_{c5}\}$ is the diagonal matrix of the parameters of the motor torque constant

$K = \text{diag}\{K_5\}$ is the diagonal matrix of the parameters of the gain of the current amplifier

$v \in \mathbb{R}^5$ is the the vector of the input voltage of the motors.

Substituting Γ , \ddot{q}_m and \dot{q}_m into Γ_m yields

$$N.K_c.K.v = (I_a.N^2 + H(q)) .\ddot{q} + (N^2.F_v + C(q, \dot{q})) .\dot{q} + g(q) \quad (1)$$

It can be deduced

$$\ddot{q} = - (I_a.N^2 + H(q))^{-1} (N^2.F_v + C(q, \dot{q})) .\dot{q} + (I_a.N^2 + H(q))^{-1} N.K_c.K.v - (I_a.N^2 + H(q))^{-1} g(q) \quad (2)$$

Let us introduce the state vector $x \in \mathbb{R}^{10}$: $x = [x_1 \ x_2]^T = [q_1 \ \dots \ q_5 \ \dot{q}_1 \ \dots \ \dot{q}_5]^T$

Substituting x into (2) yields

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = -A(q, \dot{q}).x_1 + B(q).v - G(q) \end{cases}$$

where $A(q, \dot{q}) \in \mathbb{R}^{5*5}$, $B(q) \in \mathbb{R}^{5*5}$ and $G(q) \in \mathbb{R}^5$ with

$$A(q, \dot{q}) = (I_a.N^2 + H(q))^{-1} (N^2.F_v + C(q, \dot{q}))$$

$$B(q) = (I_a.N^2 + H(q))^{-1} N.K_c.K$$

$$G(q) = (I_a.N^2 + H(q))^{-1} g(q).$$

Only the position is measured, so the state space model is defined as, (3)

$$\begin{cases} \dot{x} = \begin{bmatrix} 0 & 1 \\ 0 & -A(q, \dot{q}) \end{bmatrix} .x + \begin{bmatrix} 0_{(5*5)} \\ B(q) \end{bmatrix} .v + \begin{bmatrix} 0_{(5*1)} \\ -G(q) \end{bmatrix} \\ y = C.x = [1 \ 0] .x \end{cases} \quad (3)$$

2.2 Fault analysis

An inductive method, Failure Mode, Effects, and Critical Analysis (FMCEA) ,(Visinsky *et al.*, 1994) and (Nouredine, 1996), allows the enumeration of possible failures and their failure modes and then tabulates the effect of each failure. The standardized FMCEA chart can be used. A table includes causes of failure, system effect and recommended design adaptations to reduce the effect of each failure.

2.3 Fault modelling

Most FDI works assimilate the faults to additive terms in the state equation when the model with faults has to be designed. This term is added to the dynamic equation for dynamic faults and another to the measure equation for the sensor faults. This formulation, if it does not allow the faulty parts of the system to be stressed, has the advantage of keeping the linear properties of the model and permits the application of all linear techniques developed for Linear Time Invariant systems for the detection and isolation.

2.3.1. Sensor faults The sensor faults will affect the position values given by the optical encoders of each joint. To simulate the occurrence of a fault in the position measure, an offset will be generated on the output of the modelled system. A vector of sensor fault (measurement fault) $f_m \in \mathbb{R}^5$ multiplied by a fault distribution matrix $K_m = \text{diag}\{K_{m5}\}$ is defined in the output equation of the state space model $y = C.x + K_m.f_m$ with $f_m = [f_{m1} \ f_{m2} \ \dots \ f_{m5}]^T$

2.3.2. Dynamic faults Dynamic faults occur on the mechanical part of the robot, on the actuators and on the electronic interfaces. This type of fault will appear in the dynamic equation of the system as a vector $fd \in \mathbb{R}^5$ multiplied by a fault distribution matrix $K_d = \text{diag}\{K_{d5}\}$,

$$\dot{x} = \begin{bmatrix} 0 & 1 \\ 0 & -A(q, \dot{q}) \end{bmatrix} .x + \begin{bmatrix} 0_{(5*5)} \\ B(q) \end{bmatrix} .v + \begin{bmatrix} 0_{(5*1)} \\ -G(q) \end{bmatrix} + \begin{bmatrix} 0_{(5*5)} \\ K_d \end{bmatrix} .fd$$

with $fd = [fd_1 \ fd_2 \ \dots \ fd_5]^T$

2.3.3. Fault on joint 3 The model of dynamic faults injected into the system concerns the servoamplifier. The actuators and their associated electronic interfaces are main parts in mechatronic systems and some works still dealt with them (Wang and Daley, 1996). In this paper the injection of faults has been achieved by introducing short circuits which induce a low or high saturation of the output of some operational amplifiers. These faults and their modelling are compatible with the linearity condition capable of including an additive fault, $fd = B(q).(v_{sat} - v)$. It is obvious that these types of faults can occur on any joint but in order to illustrate the fault accommodation algorithms we will consider only fault occurrence on joint 3. This restriction does not question the whole fault accommodation procedure, but simplifies the fault detection procedure and more precisely avoids the isolation stage, as it is admitted that any detected fault will occur on joint 3.

3. STATE SPACE CONTROL AND FAULT DETECTION

The Fault detection and isolation algorithms are very closely linked to the control algorithms. Several methods are usually used and especially the exact linearization through state feedback, named computer torque method in the joint space, when the aim is control tracking. When the objective is point to point motion, a local linearization is possible around a fixed point. In this paper the specified task given to the robot in question belongs to this category.

3.1 Operating point

The operating point fixed in our application is defined by the vector of the end-effector position $P = (P_x \ P_y \ P_z)$ which establishes the end-effector coordinates in the frame R_0 , with $P_x = 0.30m, P_y = 0, P_z = 0.80m$. The end-effector orientation is imposed by 2 angles, the pitch angle, noticed q_{234} and the yaw angle, noticed q_5 with $q_{234} = 10^\circ$ and $q_5 = 0^\circ$.

The vector of joint variables is achieved thanks to the inverse kinematic model. When the desired position and orientation of the end-effector are given, this model permits the computation of a set of joint angles which will obtain the desired position of the end-effector. The equations are established by the Denavit-Hartenberg formalism, (Craig, 1989), and validated by the Symoro+ software, (Khalil and Creusot, 1997).

q_1 is obtained from the first equation of the inverse kinematic model

$$Px * \sin q_1 - Py * \cos q_1 = 0, \text{ so } q_1 = \arctan\left(\frac{Py}{Px}\right)$$

The angle q_3 is calculated by $q_3 = \arccos\left(\frac{A}{B}\right)$

$$\begin{aligned} \text{with } A &= (Z_2)^2 + (Z_1)^2 - (D_3)^2 - (D_4)^2 \\ B &= 2 * D_3 * D_4 \\ Z_1 &= P_z - R_1 + R_5 \cos(q_2 + q_3 + q_4) \\ Z_2 &= P_x \cos(q_1) + P_y \sin(q_1) \\ &\quad - R_5 \sin(q_2 + q_3 + q_4) \end{aligned}$$

According to the signs of the variables A and B, q_3 will be obtained by possible 2 solutions and only the positive value will be held.

Then q_2 is obtained by

$$\sin(q_2) = \frac{Z_1 B_1 - Z_2 B_2}{(B_2)^2 + (B_1)^2}, \cos(q_2) = \frac{Z_1 B_2 + Z_2 B_1}{(B_2)^2 + (B_1)^2}$$

so $q_2 = \arctan\left(\frac{\sin(q_2)}{\cos(q_2)}\right)$, with :

$$B_1 = D_3 + D_4 \cos(q_3), B_2 = D_4 \sin(q_3)$$

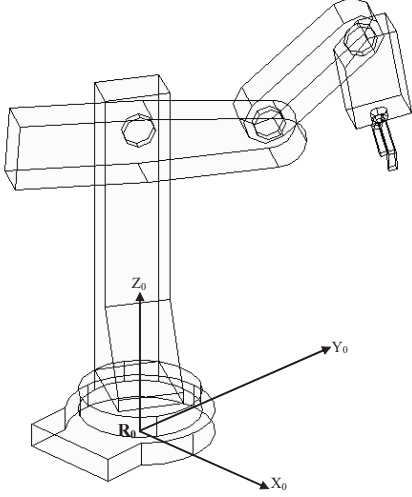


Fig. 1. Operating point

q_4 is deduced thanks to q_{234} which is a given value defined by the imposed orientation.

$$q_4 = q_{234} - q_2 - q_3$$

The vector of the joint, relative to the chosen fixed point, is finally obtained, see figure (1).

$$q_e = (q_1 \ q_2 \ q_3 \ q_4 \ q_5)^T \text{ with : } q_1 = 60^\circ$$

$$q_2 = -4^\circ, q_3 = 45^\circ, q_4 = -31^\circ \text{ and } q_5 = 0^\circ.$$

This point is validated thanks to the equation (6) which defines the workspace of the robot.

3.2 Local linearization

The robot is dedicated to operate around a fixed point which is an equilibrium point $E = (y_e, x_e, v_e)$. A local linearization around this point will be achieved. $\Phi(\dot{x}, x, v)$ and $\Psi(y, x, v)$ are written

$$\begin{cases} \Phi = \begin{bmatrix} \dot{x}_1 - x_2 \\ \dot{x}_2 + A(q, \dot{q}).x_2 - B(q).v + G(q) \end{bmatrix} = 0 \\ \Psi = y - x_1 = 0 \end{cases} \quad (4)$$

Let us define the following variables : $\tilde{x} = \begin{pmatrix} \tilde{x}_1 \\ \tilde{x}_2 \end{pmatrix} = \begin{pmatrix} x_1 - x_{e1} \\ x_2 - x_{e2} \end{pmatrix}$, $\tilde{y} = y - y_e$ and $\tilde{v} = v - v_e$ where y_e and v_e are respectively the output vector and the input voltage vector at the equilibrium point. This last vector is obtained by substituting $\dot{q}_i = \ddot{q}_i = 0$ into equation (1), thus : $v_e = [N.K_c.K]^{-1}.g(q)$ which leads to $v_e = (v_{1e} \ v_{2e} \ v_{3e} \ v_{4e} \ v_{5e})^T$

The linear model is obtained by

$$\begin{cases} \dot{\tilde{x}} = A_{lin}.\tilde{x}(t) + B_{lin}.\tilde{v}(t) \\ \tilde{y}(t) = C_{lin}.\tilde{x}(t) \end{cases} \quad (5)$$

$$\begin{aligned} \text{with } A_{lin} &= - \left[\frac{\partial \Phi(\dot{x}, x, v)}{\partial \dot{x}} \Big|_E \right]^{-1} \cdot \frac{\partial \Phi(\dot{x}, x, v)}{\partial x} \Big|_E \\ B_{lin} &= - \left[\frac{\partial \Phi(\dot{x}, x, v)}{\partial \dot{x}} \Big|_E \right]^{-1} \cdot \frac{\partial \Phi(\dot{x}, x, v)}{\partial v} \Big|_E \\ C_{lin} &= - \left[\frac{\partial \Psi(\dot{x}, x, v)}{\partial \dot{x}} \Big|_E \right]^{-1} \cdot \frac{\partial \Psi(\dot{x}, x, v)}{\partial x} \Big|_E. \end{aligned}$$

where Φ and Ψ are given in (4).

3.3 State space control and fault detection

To obtain a position error which leads asymptotically to zero, a state feedback is designed. The closed-loop poles are defined by a pole placement design method. First the controllability-canonical

$$\text{form of (5) is achieved by } \begin{cases} \dot{\tilde{x}}_c = A_c \tilde{x}_c + \bar{B}_c \tilde{v}_c \\ \tilde{y}_c = C_c \tilde{x}_c \end{cases}$$

with A_c, \bar{B}_c and C_c the appropriate matrix are calculated thanks to Luenberger algorithms

The state vector \tilde{x}_c is obtained thanks to the transformation T_c such as $\tilde{x}_c = T_c \tilde{x}$.

A proportional state space controller enables one to define : $v = v_e - K_v.\tilde{x}_2 - K_p.\tilde{x}_1$ where K_v and $K_p \in \mathbb{R}^{5 \times 5}$ and represents the state feedback gain.

Only the angular position vector x_1 is measured and thus it is necessary to estimate the angular velocity vector x_2 . The gain matrix L is obtained from the observability canonical form. For the stability of the observer and in order that the error converges to zero very fast, the eigenvalues of $A - LC$ must be sufficiently negative

$$\begin{cases} \dot{\tilde{x}}_o = A_o \tilde{x}_o + B_o \tilde{v}_o \\ \tilde{y}_o = C_o \tilde{x}_o \end{cases}$$

where A_o, B_o and C_o are appropriate matrix.

Using an observer for the control has enabled us to generate a position residual.

The residual generation using observers is one of the main model based methods. A large number of surveys can be found in the literature and describes all the possibilities and performances of these techniques, (Garcia and Franck, 1997) and (Isermann, 1997).

A full order observer is thus designed to estimate the positions. These are measured and permits the creation of residuals. The control architecture including the detection stage is described in figure 2. The observer has the form

$$\begin{cases} \dot{\hat{x}} = (A_{lin} - L.C_{lin}) \hat{x} + B_{lin}.\tilde{v} + L\tilde{y} \\ \hat{y} = C\hat{x} \end{cases}$$

The residual r is deduced by $r = y - \hat{y} = C(x - \hat{y}) = Ce$

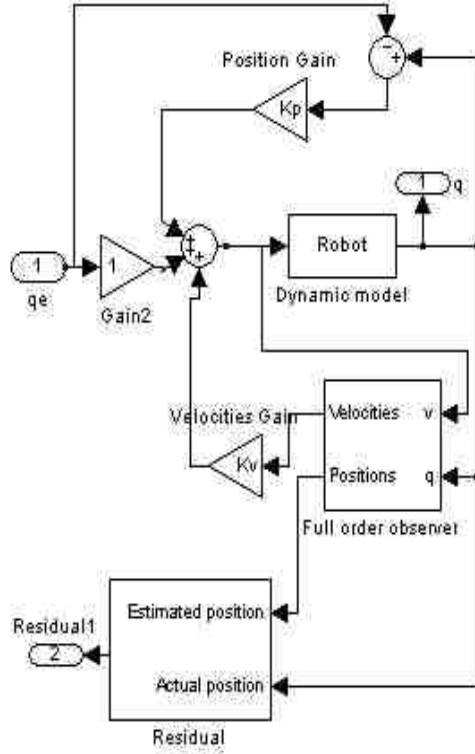


Fig. 2. Control and detection scheme

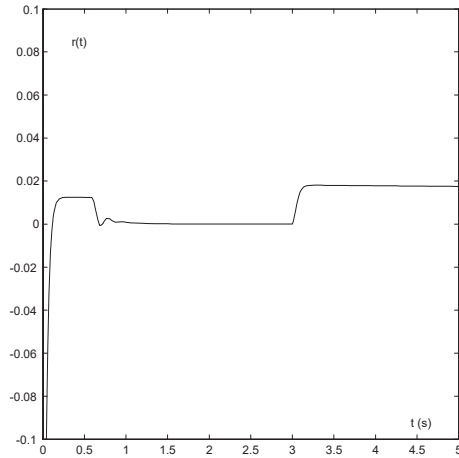


Fig. 3. Residual due to a dynamic fault

3.3.1. Simulation and residual behavior analysis

As mentioned in paragraph 2.3.3, a saturation at +5V, is implemented à t=3 s, a significant residual value is deduced which expresses the fault occurrence, see figure 3. The residual has also a non-negligible value during the end-effector motion. This phase lasts about 0.75s. From 0.75s to 3s, the residual $r(t)$ is almost nil due to a normal mode.

A decision stage, which can be applied in the case where the signals are little noisy, is made by thresholding the residual $r(t)$. Remembering that this detection is highly facilitated by the fact that only faults on joint 3 are taken into account. The problem will be more complicated if all the

possible faults, on the 5 joints are considered, and in particular the problem of locating the fault. A scheme of bank of observers coupled with some unknown input observers are generally necessary to obtain structural residuals and allow fault detection and isolation (Patton and Chen, 1997), (Noureddine *et al.*, 1994).

4. FAULT ACCOMMODATION

Due to their mechanical structure, some robots can have the advantage of being kinematically redundant. In the general case, the position and the orientation of the end-effector within the workspace necessitates 6 degrees of freedom. Any kinematic redundancy would require a seventh joint for example. For the robot in question and although it has only 5 joints, that is to say limited in its orientation, the mechanical structure implies that joints 2, 3 and 4 are moving in the same plan, thus creating a redundancy for the positioning of the end-effector. This structural redundancy can be used to create fault tolerant algorithms which could be used to provide alternative configurations in failure situations. This redundancy has the great advantage of not requiring any supplementary joint and thus not complexifying the mechanical structure (Groom *et al.*, 1999), (Chen *et al.*, 2003), (English and Maciejewski, 2000). It is well known that the more mechanical components we add, the more we fragilise the whole system. In the same way, any failure of the joint of the robot can lead to both a loss of accessibility in the task space or a loss of orientation capability, but a trajectory reconfiguration will enable a working in a more limited form. Any robot manipulator can have some redundancy capability for a given task. In order to study these capabilities it is very important to identify the workspace of the robot. The work envelope is a given specification by

$$\begin{aligned} 0 < q_1 < 180^\circ, & -15^\circ < q_2 < 80^\circ \\ -105^\circ < q_3 < 105^\circ, & -35^\circ < q_4 < 195^\circ \\ -360^\circ < q_5 < 360^\circ \end{aligned} \quad (6)$$

4.1 Failure of joint 3

To illustrate the fault accommodation procedure, joint 3 is considered as locked in its initial position. The new and reduced workspace corresponds to a joint situation where joint 3 is maintained to 0. To calculate the alternative joint configuration which should permit the operating point to be reached, we had to calculate once again the inverse kinematic model with $q_3 = 0$. Remembering that the operating point is given by a position specified by $P_x = 0.30m, P_y = 0, P_z = 0.80m$ and an

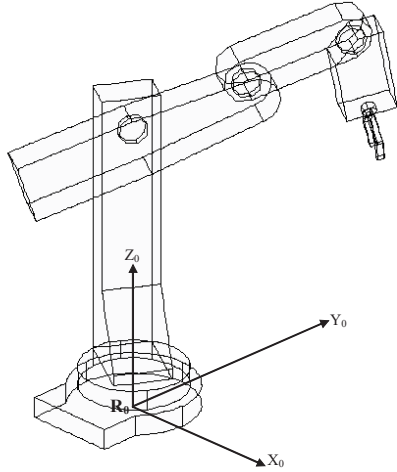


Fig. 4. Redundant joint configuration

orientation specified by $q_{24} = 10^\circ$ and $q_5 = 0^\circ$. The angles are given by

$$q_1 = \arctan \frac{P_y}{P_x} = 60^\circ.$$

$$q_2 = \arctan \frac{S_2}{C_2} \text{ with } S_2 = \frac{B_2}{D_3 + D_4} \text{ and } C_2 = \frac{B_4}{B_3}$$

where :

$$B_2 = P_z - R_1 + R_5 \cos(q_{24})$$

$$B_3 = (D_3 + D_4) \cos(q_1)$$

$$B_4 = P_x - R_5 * \cos(q_1) * \sin(q_{24})$$

and finally : $q_2 = 18^\circ$

q_4 is obtained by $q_4 = q_{24} - q_2 = -8^\circ$

The alternative joint configuration is $q_1 = 60^\circ$,

$q_2 = 18^\circ, q_3 = 0, q_4 = -8^\circ$ and $q_5 = 0$, and is shown by figure 4.

5. CONCLUSION

This work is a contribution to illustrate the fault tolerance concepts in robotics. All the steps, from analysis phase, to the fault detection stage and the control, are briefly presented to let a more important space for the development of the fault accommodation phase. The latter is generally less developed than the FDI stage. The considered trajectory is a point to point motion and the implemented control includes a local linearization around the fixed point. A dynamic fault and a measurement fault are introduced to validate the detection algorithms. The fault accommodation is thus considered and is based on the kinematic redundancy principle. The calculation of the inverse kinematic model, under some conditions of joint failure, permits the generation of an alternative trajectory to ensure the robot continues to function.

REFERENCES

- Canudas, C., B. Siciliano and G. Bastin (1997). *Theory of robot*. Springer Verlag.
- Chen, Y., J. McInroy and Y. Yi (2003). Optimal, fault-tolerant mappings to achieve secondary goals without compromising primary performance. *IEEE Trans. on Robotics and Automation* **19**(4), 680–691.
- Craig, J.J. (1989). *Introduction to robotics*. Addison-Wesley.
- English, J.D. and A.A. Maciejewski (2000). Measuring and reducing the euclidean-space effects of robotics joint failures. *IEEE Trans. on Robotics and Automation* **16**(1), 20–28.
- Filaretov, V.F., M.K. Vukobratovic and A.N. Zhibabob (2002). Parity relation approach to fault diagnosis in manipulation robots. *Mechatronics* **12**(8), 999–1010.
- Garcia, E.A. and P.M. Franck (1997). Deterministic non linear observer-based approaches to fault diagnosis : A survey. *Control Eng. Practice* **5**(5), 663–670.
- Groom, K.N., A.A. Maciejewski and V. Balakrishnan (1999). Real-time failure-tolerant control of kinematically redundant manipulators. *IEEE Trans. on Robotics and Automation* **15**(6), 1109–1116.
- Isermann, R. (1997). Supervision, fault detection and fault-diagnosis methods-an introduction. *Control Eng. Practice* **5**(5), 639–652.
- Khalil, W. and D. Creusot (1997). Symoro+ : A system for the symbolic modelling robot. *Robotica* **15**, 153–161.
- Noureddine, F. (1996). Conception d'un outil logiciel graphique pour l'analyse comportementale d'un robot en mode dégradé. *Revue Internationale de CFAO et d'Informatique Graphique* **11**, 199–214.
- Noureddine, F., M.S. Lazeregue and D. Noyes (1994). Fault detection-location in robots using scheme of observers. In: *Safeprocess'94*. Helsinki.
- Patton, R.J. and J. Chen (1997). Observer-based fault detection and isolation: robustness and applications. *Control Eng. Practice* **5**(5), 639–652.
- Visinsky, M.L., J.R. Cavallar and I.D. Walker (1994). Robotic fault detection and fault tolerance : A survey. *Reliability Engineering and System Safety* **46**, 139–158.
- Visinsky, M.L., J.R. Cavallaro and I.D. Walker (1995). A dynamic fault tolerance framework for remote robots. *IEEE Trans. on Robotics and Automation* **11**(4), 477–489.
- Wang, H. and S. Daley (1996). Actuator fault diagnosis : An adaptive observer-based technique. *IEEE Trans. on Automatic Control* **41**(7), 1073–1078.